Operational Topic

This paper outlines the design process followed by an interdisciplinary undergraduate team and presents their design of an affordable Geiger-Muller survey instrument controlled by a smartphone application with mapping capability suitable for use in high school outreach programs.

Design of a Do-It-Yourself Geiger-Muller Counter With Smartphone Mapping Application

Jeffery B. Xiao, James M. Seekamp, Long Kiu Chung, Issa N. El-Amir, Kai C. Schiefer, David J. Trimas, Regina Tuey, and Kimberlee J. Kearfott[1]

Abstract: *Legacy Geiger-Muller (GM) survey meters recovered from fallout shelters have been used by several nuclear scientific societies as part of high school outreach programs. A donated antique instrument helps teachers demonstrate radiological principles, but fails to develop student's electronics skills, generate excitement for nuclear careers, or provide individuals with their own devices to explore the radioactive planet. A simple, affordable GM survey meter built by each student would increase direct engagement while providing hands-on experience with circuit-building, soldering, and computer programming. The inclusion of an affordable single-board computer as a component in the survey meter would enable students to tackle more various computer science and electronics projects, thereby potentially recruiting more students into technology and engineering. This paper details the challenges faced by an interdisciplinary undergraduate team designing an easy-to-assemble smart GM survey meter. Their iterative research, design, and testing process included modification to a basic circuit to enable use of different tube types, component cost reduction, application development, and data communication. The ultimate product of the team's efforts, a survey meter with affordable components and a smartphone application capable of creating radiation maps, is detailed in full. Health Phys. 116(6):000–000; 2019*

Key words: *operational topics; detector, radiation; education, health physics; survey meter*

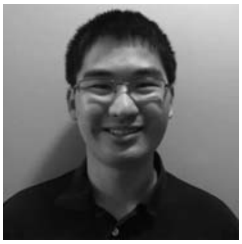
INTRODUCTION

Since as early as the 1950s and continuing through the 1970s (Powers 1950; Folger and Meeks 1957; Shilling 1960; Kuhn 1966; Moeller 1971; Deutsch and Whitney 1974; Csik et al. 1979), there has been concern regarding a lack of qualified personnel in the radiation health and nuclear power fields. Early on, during the rapid growth and development of applied nuclear sciences, concerns were expressed about the technical background of entering professionals. Over the following decades and until the present day, job opportunities periodically cycled in both nuclear engineering and health physics. Throughout this time, concerns over the number of students entering nuclear fields have persisted (Johnson 1982, 1991; Stevenson 1984; Was and Martin 2000; Kaiser 2002; Wogman et al. 2005; Hansen 2008; Xue and Larson 2015). Many have speculated that the applied nuclear sciences fields are neither promising nor interesting enough for students, or perhaps this reflects lessened interest in science and engineering in general.

To combat this, the American Nuclear Society (ANS) teacher workshop provides high school teachers access to educational materials, radiation detectors, and exempt radiation sources with the hope of engaging their students in learning about radiation and nuclear science (CNSTI 2018). The radiation detectors offered through the program typically included antique detectors scavenged from fallout shelters. These detectors, like vintage direct reading ion chambers from the same source (Bergen et al. 2010), are very reliable once a specific capacitor with a limited shelf life has been replaced. This ANS program achieves the goal of supplying teachers with teaching materials and instrumentation suitable for demonstrations. Educators are

---

[1] *Department of Nuclear Engineering and Radiological Sciences, University of Michigan, 2355 Bonisteel Boulevard, Ann Arbor, Michigan 48109-2104*



The authors declare no conflicts of interest.

Jeffery Xiao is currently a first-year master's student studying medical physics at Duke University. He is a full-time student who is exploring options for part time employment and research work. His goal is to apply for residency pending graduation in 2020 and pursue a career in the clinical medical physics field.

thus readily enabled to teach their students about nuclear science, which can ignite career interest while promoting fundamental radiation literacy.

The Health Physics Society (HPS) has a program very similar to that of the ANS, and includes a website featuring classroom resources (HPS 2018). While some local HPS chapters provide legacy fallout shelter GM survey meters to teachers in collaboration with the ANS, a radiation instrument kit loan program is also offered by the HPS. This commercially available kit features a standard GM detector with a computer interface and software (Vernier 2018). The computer interface and software are platform-dependent, so in-stock versions through the HPS may become quickly obsolete, and not necessarily suitable for Windows, iOS, Android, or other platforms. The current recommended demonstration kit for Windows computers has a list price of ~$600.

For over 25 years, more active hands-on teaching methods have been promoted as superior to traditional lecture- and demonstration-based classes, especially for technical subjects such as science and engineering at the high school and college levels (Felder and Silverman 1988; Wellington 1990; Stohr-Hunt 1996; Carlson and Sullivan 1999; Mestre 2001; Gibson and Chase 2002; Markowitz 2004; Klahr et al. 2007; Corter et al. 2011; Singer and Smith 2013). In addition, students increasingly prefer an active learning process (Wilson 2004; Roehl et al. 2013). The current generation prefers doing and thinking, rather than listening and thinking. Furthermore, when projects can be related to everyday experiences, they are even more powerful (Hulleman and Harackiewicz 2009). This would be the case when students would have the ability to continue hands-on exploration beyond the science classroom or laboratory.

Putting a radiation survey meter in the hands of each student, rather than simply for their teacher to use during class periods, would enable them to continue exploring radiation beyond the experience of a simple demonstration. Having each student construct their own detector provides an added engineering-rich experience compared to that of simply having access to radiation detection equipment. Radiation detectors, based on deflections of a needle in response to static electrical charges, may be created from simple household materials (McDonald et al. 2004). Unfortunately, those detectors do not work well in conditions of high humidity, are extremely insensitive, and suffer significant limitations in terms of portability and ability to examine samples for radioactive content. They also lack the appeal of a modern digital system. A computer- or cellphone-enabled system achieves a more modern appeal than that of a vintage fallout shelter unit or a detector made from a soda can.

A do-it-yourself (DIY) Geiger-Mueller (GM) survey meter, while it may be time consuming for students to build, would ultimately teach them more than a demonstration, provide a useful instrument, and be more effective for a generation of students with hands-on learning preferences. During assembly of such a device, students will acquire valuable skills. Circuit design, soldering, and a basic understanding of computer science are universally essential for the rising technologically-oriented student. Having the advantage of being introduced to such important fields before reaching college will give students a beneficial head start. Hopefully, a young person's positive memory of their first engineering project will draw them towards the nuclear sciences and radiation detection fields.

This paper summarizes the design process and final DIY GM product of an undergraduate team, consisting of students studying mechanical, electrical, and nuclear engineering as well as computer science and art. The DIY GM design and this paper itself represent an outreach effort by college students resulting in increased interests in health physics and an understanding of radiation measurements.

METHODS

Background information

GM tubes are gas-filled detectors operated at high voltages used to count gamma and beta radiation-induced interactions. The high voltage runs along the length of the tube in a positively charged anode wire in the center of the tube, while the outer shell is the negatively charged cathode. This charge difference causes a large electric field around the anode wire. Surrounding this wire, inside the airtight cathode tube, is a fill gas. This fill gas is typically composed of either one or a mix of two noble gases in addition to a quench gas.

Beta radiation that enters the GM tube is accelerated by the strong electric field and collides with the neutral noble gas molecules, causing excitation and ionization. These freed electrons continue the pattern in what is known as a Townsend avalanche (Knoll 2010), with the noble gases providing ion pairs and the quench gas absorbing the photons released as excited molecules transition to their ground states. All the freed electrons build and collect along the anode wire while the positively charged ions they originated from slowly move towards the cathode. This movement of charge is what causes the output current pulses that are counted. Photons cause the same effect to create a pulse with the extra step that the photon must collide and free an electron from one of the atoms in the fill gas before it can be detected.

If a higher operating voltage is employed, a GM tube will have a greater electric field between the anode wire and the cathode tube, resulting in many more ion pairs and excited molecules. The excited molecules de-excite, creating more photons, causing more avalanches. As these avalanches build, the newly freed electrons must drift further and further along the anode until the entire wire is surrounded by the oppositely charged ions. The avalanches eventually terminate due to the comparatively slow movement of the positive ions; as the positive ion concentration builds, they begin to reduce the internal electric field of the tube, halting further avalanches.

Through this method of avalanche buildup and natural termination, so long as a fixed voltage above the minimum is applied to the GM tube, the response pulse from the GM tube will always be constant. For this reason, GM tubes have a constant signal response regardless of the initial particle type or energy. The loss of these details allows for GM tubes to work as detectors for a total number of radiation events, but not for radiation spectroscopy.

The increased simplicity, low cost, and ease of use is the reason that GM tubes were selected for this outreach program over other detectors such as scintillators. Similar ideas using silicon pin diodes (Williams 2014) were also passed over due to their lower sensitivity when compared to GM detectors. Through the construction and operation of a simple GM counter, younger students will be exposed to science, technology, engineering, and mathematical (STEM) topics, including but not limited to radiation detection, circuit building, soldering, and computer programming.

GM tube selection

The availability of a wide variety of GM tubes makes selecting a tube important. Different tubes will have different reliabilities, sensitivities, operational requirements, and costs. The most important requirement is the operational voltage, which typically is 350 to 1,000 V. Additionally, GM tubes also come in different shapes and sizes, ranging from small pins, to pancake shaped probes, to classic cylindrical tubes. The GM tubes used for this project, optimally operated at 900 V, were Victoreen model 6993 GM tubes, shown in Fig. 1, salvaged from the legacy CD V-700 fallout shelter survey instruments (Victoreen Instruments Company, E 67th St & Central Ave, Cleveland, OH 44103). Due to the age of these tubes, a series of tests had to be done to sort out the ones that worked from the ones that had succumbed to time. While not all tubes were salvageable, roughly 70% of the tubes were, and their free nature is of great benefit for maintaining low outreach program costs. The purpose of this outreach program is to introduce and educate, so professional quality equipment is not necessary. It is also important to note that the GM tubes represent a component of the system which, in fact, have not changed much in design for nearly half a century.

Circuit design iterations

The circuit design process began with a prebuilt board (Giametti 2018) shown in Fig. 2a as a reference. It was soon found, however, that 900 V was higher than typical for many modern or Russian surplus tubes for which the prebuilt board was designed. Because of this, a revised custom circuit design was needed. This process would take several iterations, with the circuit prototypes constructed on breadboards (Half Size Breadboard, Adafruit Industries, New York, United States, 150 Varick Street, NY 10013).

The initial challenge faced was the general lack of experience designing and building circuits among the interdisciplinary undergraduate team members involved in the project. Due to these gaps in knowledge, early designs attempted to imitate the circuit board design of the marketed board by tracing how the various wires connected to components. This was done to investigate which sections of the circuit were meant to power the GM tube and which parts were for the other components on a detector, such as a speaker or LED light. The hope was to change and improve the design to achieve the voltages the model 6993 tubes required. This first step was aided by the manufacturing group providing their circuit information for all to view on their website, given in the references.

This very first circuit design, shown in Fig. 2b, was disorganized and messy, and done without making full use of breadboards.

FIG. 1. Sample Geiger Muller tubes salvaged from legacy Victoreen CDV 700 survey instruments recovered from fallout shelters.

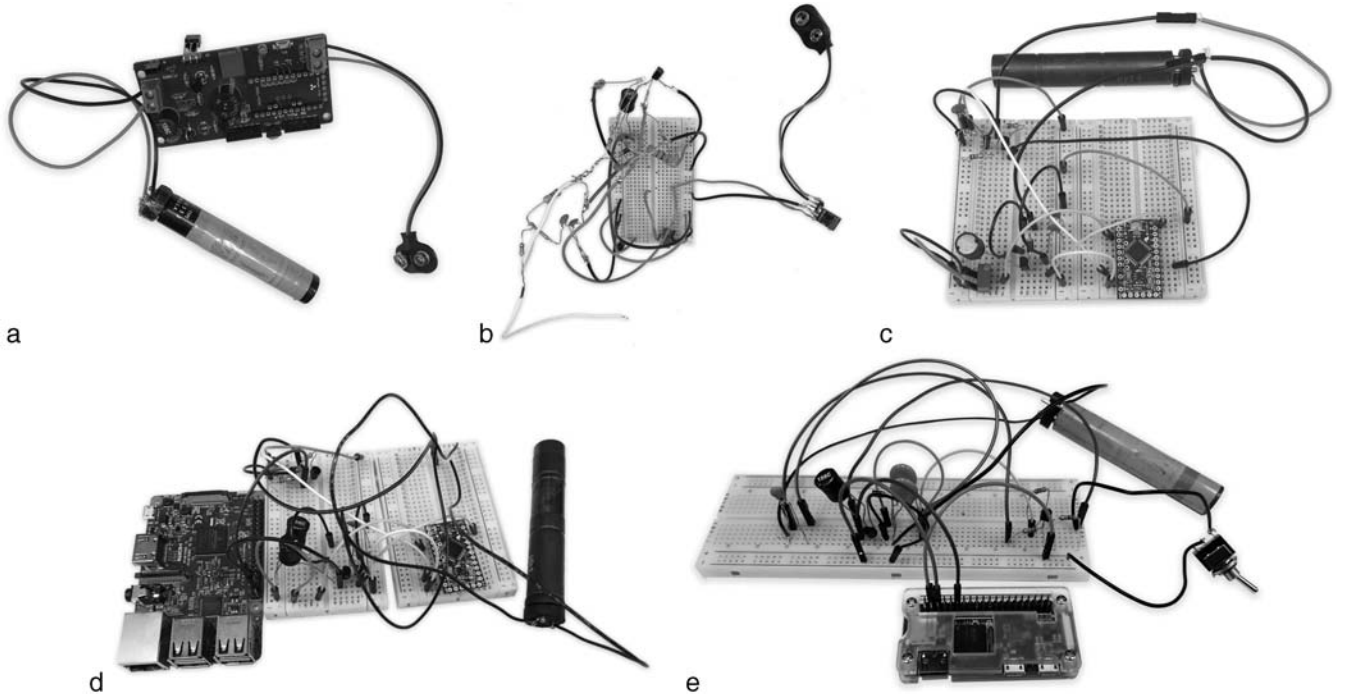


FIG. 2. Etiology of purchased kit and bread-boarded designs of the detector, specifically: a) Purchased design, which did not provide ample voltage for the fallout shelter tubes; b) Unsuccessful and poorly organized first prototype circuit attempt powered by 9 V battery; c) Intermediate prototype circuit including an insufficient voltage boost with an Arduino microcontroller and lacking data display mechanism; d) First successful prototype employing both Raspberry Pi and Arduino microcontroller, capable of supplying ample voltage and digital data export; and e) Final working prototype re-designed to reduce component costs.

Ultimately the first attempt had to be scrapped in favor of a better planned design. Further study was done to piece together the necessary sections of the purchased circuit. Eventually, a minimal circuit diagram was decided on and drawn based on what was directly between the battery and the tube, and between the tube and the microcontroller.

With the purchase of breadboards, the second iteration could be built in a far more organized fashion, shown in Fig. 2c. From the beginning, it was clear that there had to be some method to increase the voltage from the battery to the necessary levels for the GM tube. By studying the circuit diagram, it was determined that a microcontroller, the Mini Arduino Pro (Arduino LLC, 72 Oak St #4, Somerville, MA 02143-4034), could provide the voltage amplification. Such microcontrollers provide a simple way of increasing the voltage from batteries, which use direct current (DC), to the required levels for the fallout shelter GM tubes. The addition of the microcontroller also meant that the original voltage from the 9 V battery had to be dropped to the maximum 5 V allowed, which was accomplished by adding a voltage regulator to the circuit. With the simplified circuit built, testing revealed that the voltage was still too low to properly use the model 6993 GM tube.

Soon after, the design also shifted to incorporate a Raspberry Pi 3B (Adafruit Industries, 150 Varick Street, New York, NY 10013) to display any received data and test the circuit more easily. Once the Raspberry Pi was formatted, a short Python code was written to receive pulse signals from the GM tube, count them per set time interval, and display the total counts per time interval continuously. The circuit and code were then all tested with a more modern GM detector (Model 7317, LND, Inc., 3230 Lawson Boulevard, Oceanside, NY 11572) with a lower voltage requirement to confirm that both the circuit and the code worked when a radioactive source was placed nearby.

With the circuit confirmed to be working and code outputting detected data from the GM counter with more modern circuitry and controls, the process to convert the circuit and boost the voltage to the necessary levels became more streamlined. Through an iterative correction process on the voltage boost section of the circuit, the voltage was slowly brought up to the operational voltages of the model 6993 GM tube. The final voltage achieved on the breadboard was around 910 V with small amounts of fluctuation caused by changes in the inductance and resistance of the circuit, shown in Fig. 2d. The final iteration, shown in Fig. 2e, saw the removal of the microcontroller from the design since the Raspberry Pi accomplished everything that the microcontroller was being used for.

Printed circuit board (PCB) design iteration

Once completed, the working circuit diagram was drawn using electronic design automation software (Eagle, Autodesk, Inc., 111 McInnis Parkway, San Rafael, CA 94903). The same program was also used to then design and lay out the custom printed circuit board (PCB). This PCB design also went through several iterations as mistakes were found and changes were made. The first three iterations were manufactured by one group, which had a particularly rapid turn-around time and low cost for small number of boards (OSH Park LLC, 311 B Avenue # B, Lake Oswego, OR 97034). Once debugged, large quantities were iterated using an international PCB manufacturer with better quantity pricing (Quickturn PCB, 301-1007, Bucheon Techno Park, SsangYong 3-Cha, Seokcheon-ro 397, Ojung-gu, Bucheon-city, Kyunggi-do, South Korea 14449). Upon receiving and testing the first iteration, shown in Fig. 3a, changes were immediately made due to a series of vias, the small holes where wires are attached and soldered, being too small for the Raspberry Pi's 40 general purpose input/output (GPIO) pins. The second iteration, shown in Fig. 3b, included slots for the Arduino, but had faulty ground wiring.

The resulting third set of boards were improperly cut, shown in Fig. 3c, but worked with a surprising twist. Instead of the 910 V that was measured during testing with the breadboard, the PCB had an output of up to 1,000 V. Investigation revealed that breadboards may affect inductance and capacitance of a circuit, and the use of high voltages and frequencies on breadboards should be avoided (Altium 2018). The higher voltage, when tested, did not seem to affect the GM tubes, however, so no

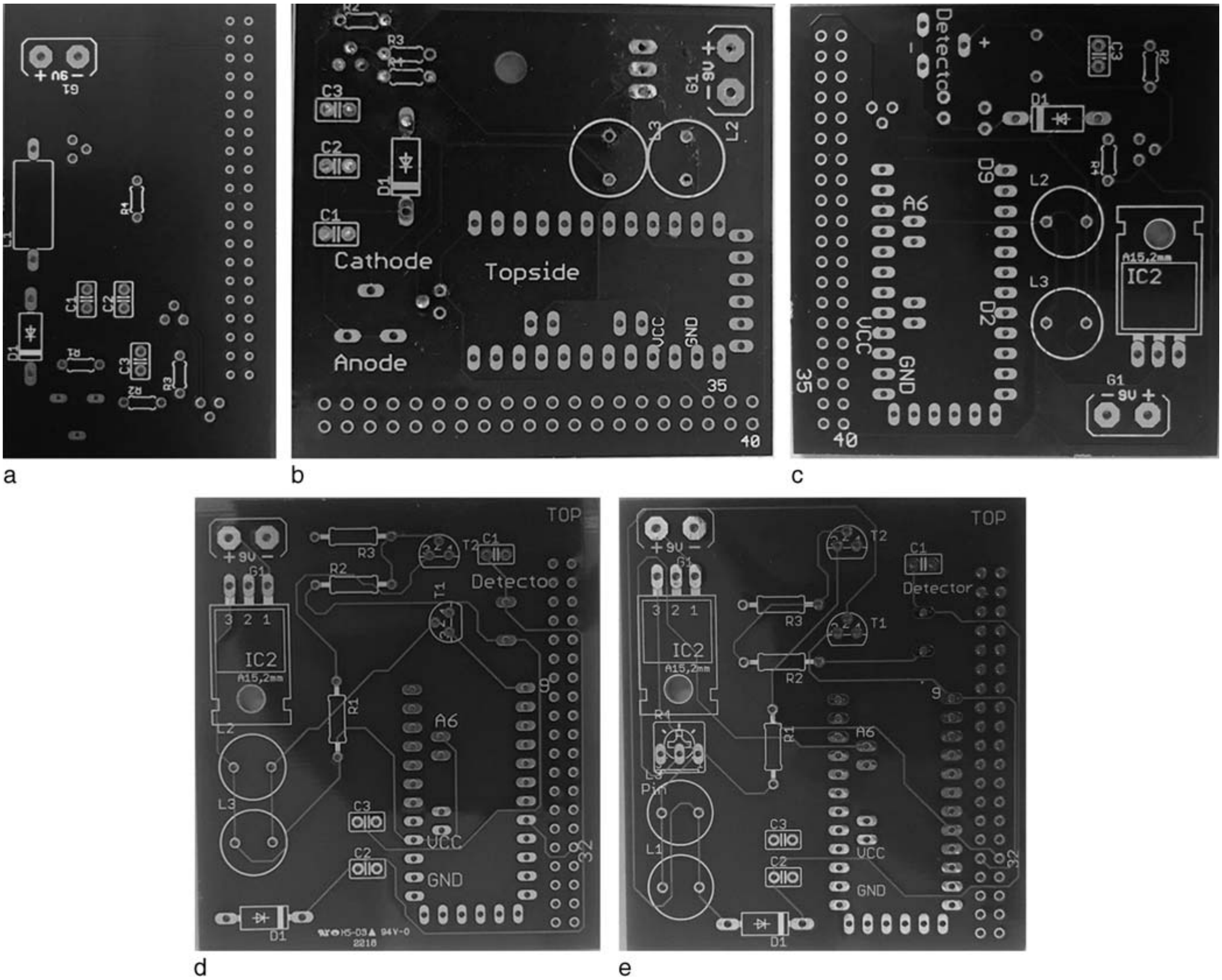


FIG. 3. Evolution of the circuit illustrated by printed circuit board (PCB) showing a) First attempt with 40 pin holes on the right too small to accommodate the Raspberry Pi GPIO pins; b) Modified PCB design with improper ground wiring; c) Functional PCB with Arduino included; d) Revised design with greater separation of components to ease assembly; e) Modified design with a potentiometer added to adjust the voltage.

immediate changes were made. Following this iteration, an instruction manual for assembling the complete detector was written for students, which would be edited appropriately with each following iteration. During testing of the instruction manual by students uninvolved with the project, it was decided to make the board slightly larger to space out some of the components and simplify the soldering process for students. The result is shown in Fig. 3d.

Between the fourth and fifth board iterations, a thorough investigation of the Arduino microcontroller code was conducted (Giametti 2018). The result of this investigation was the addition of a slot for a potentiometer into the design that allowed for easy adjustment of the voltage output should students wish to change to a different GM tube in the future, shown in Fig. 3e. This change would be short-lived, however, as, soon after, members of the team realized that the Raspberry Pi already could easily do the Arduino's job within the circuit. Through some more learning and testing, the sixth iteration of the board saw the Arduino microcontroller, 5 V step-up/step-down regulator, the slot for the potentiometer, the 9 V battery pack, and one inductor removed, resulting in a reduction in price by about 20% from the original $90 cost. This design, shown in Fig. 4a as a schematic, marks the final iteration. Fig. 4b and c, respectively, show the front and back of the PCB manufactured from this design.

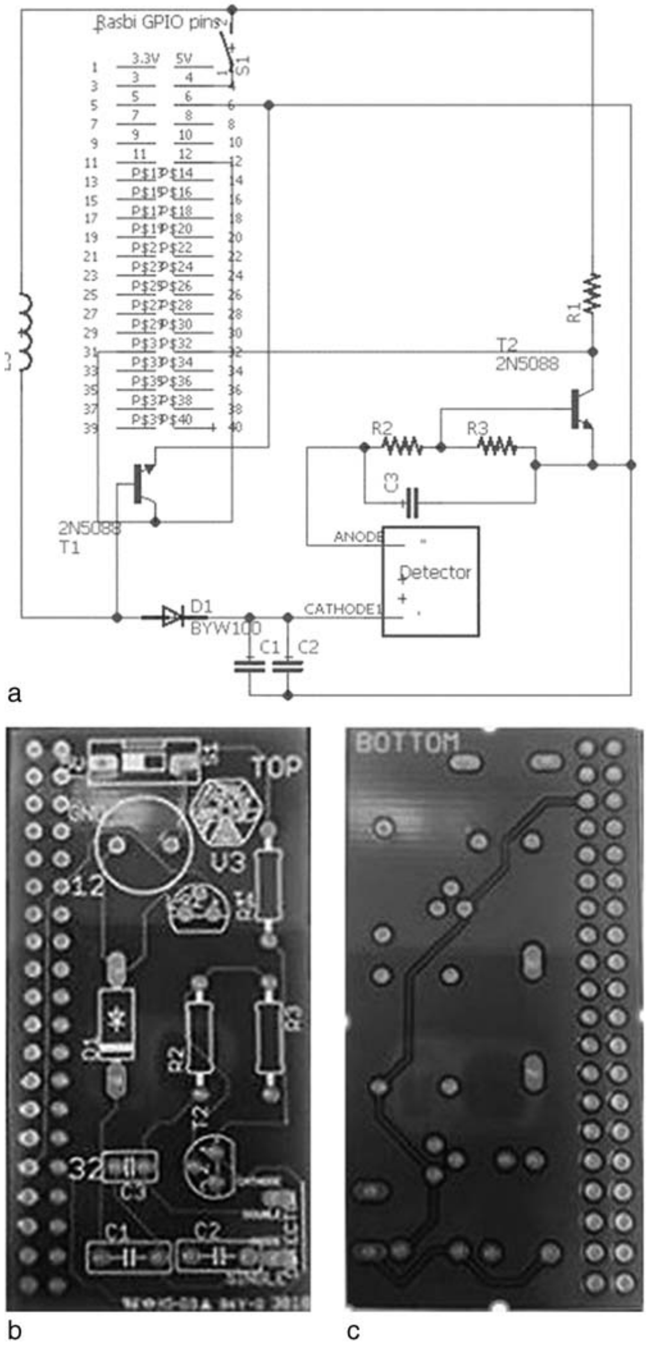


FIG. 4. Final design of circuitry for the affordable, intelligent, easy-to-assemble Geiger Muller survey meter, illustrated as a) diagram showing electronics components and Raspberry Pi connections; b) front side of final printed circuit board (PCB) having dimensions of 5.33 cm × 2.97 cm; and c) back side of final PCB.

Casing iterations

To generate designs for the container of the detector, a morphological chart (Haik 2003) was used. As shown in Table 1, subfunctions of the container were first listed, which included securing the smartphone, attaching the GM tube, and protecting the internal components. They were then charted with possible subsolutions in a grid-format, generating 4 × 4 × 4 = 64 designs. Finally, three feasible combinations, labeled A, B, and C, were identified as design candidates for further analysis. For visualization and facilitating communication, detailed sketches of design candidate A, B, and C were drawn in Fig. 5a–c, respectively.

To evaluate the design candidates, a numerical evaluation matrix (Holtzapple and Reece 2002) was used. Table 2 summarized the assigned weights to the design requirements based on their relative importance. Each candidate was then scored for each criterion under a consistent rubric. A final score

Table 1. Morphological chart for detector case design.

		Sub-functions		
		Secure phone	Attach tube	Protect components
Solutions	Solution 1	Elastic band (C)	None (A)	Injection-molded box (B)
	Solution 2	Box	Velcro (C)	Plastic ready-made box (C)
	Solution 3	Flexible net	Rings (B)	Metal ready-made box
	Solution 4	Length-adjusting clip (A) (B)	Semi-circular support	Cardboard (A)

was then obtained by multiplying each score with the weighting factor and summing each candidate's weighted scores. Manufacturing design candidate B was first attempted since it scored the highest in the numerical evaluation matrix. A simple box design was first three-dimensionally (3D)-printed for proof-of-concept. This design has a sliding cover, a cuboid containing space, and a hole for wires to go through. This design is inadequate as it did not contain most features decided for design candidate B.

To recreate design candidate B, a model was constructed using 3D computer-aided design (CAD) software (SolidWorks, Dassault Systèmes SolidWorks Corporation, 175 Wyman Street, Waltham, MA 02451) according to the sketch in Fig. 5b. This design includes rings to secure the GM tube, a semi-circular opening for wires to go through, a rectangular hole for the on/off switch, as well as simple extrusions to separate the battery from the Raspberry Pi board. Upon further research into the injection molding process however, it was soon realized that the design was not mold–friendly and is unsuitable for mass production. A change in design was needed. The third iteration included an updated CAD model of the prototype, containing features essential to injection molding, such as draft angles, radii, and smooth corners. A living hinge was also added in hopes of producing the entire part in one cycle.

During the construction of the third prototype, it was realized that if each half of the container was completely identical, the mold cost would drastically decrease as only one mold would be needed. The design was modelled in the CAD software and 3D-printed for testing. Clips were added to secure the two halves, and the holes were moved from the side to the top to reduce the number of side pins. The size of the design was modelled after an Android tablet, as a larger-than-needed dimension would essentially accommodate smartphones of all sizes. Upon physical inspection of the 3D-printed prototype, however, the container was found to not be ergonomic and user-friendly. The size was thus decided to be modelled after typical smartphones in later prototypes.

Moving to the shrunken version, the top-face dimension of the fifth prototype was reduced and sized to an Apple iPhone X, the largest iPhone model at the time (Apple 2018). The internal components were decided to be placed vertically to give more leeway in space, thus increasing the container's vertical dimension. To make the container even more ergonomic and user-friendly, the internal components were placed horizontally again, with tighter tolerances and spaces in-between. Holes were added to secure the Raspberry Pi with nuts and bolts, while the holes for wires and switches were moved to the side. Mortises were also added to secure the battery pack. Finally, a new latch design was implemented to

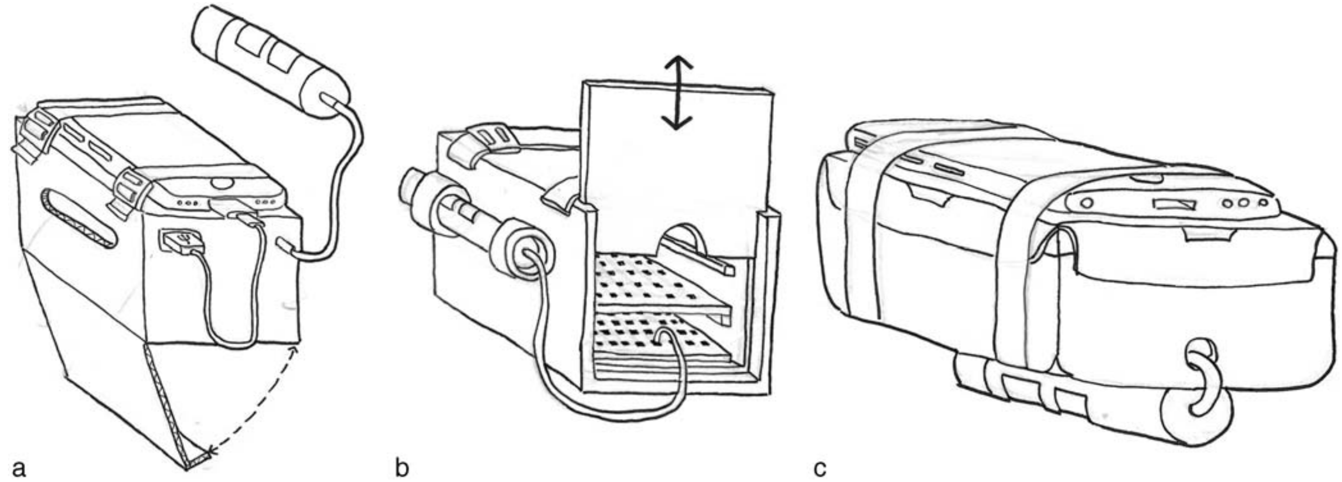


FIG. 5. Designer's sketches of conceptual case designs, including a) Candidate A has a cuboid containing space made out of cardboard, and length-adjusting clips to secure the smartphone; b) Candidate B includes rings to secure the Geiger-Mueller tube, length-adjusting clips to secure the smartphone, and is intended to be injection-molded; and c) Candidate C is made out of a ready-made plastic box, with elastic band to secure the smartphone and Velcro to secure the Geiger-Mueller tube.

Table 2. Numerical evaluation matrix for design candidates A, B, and C.

Property	Weighting factor	Score	Weighted score	Score	Weighted score	Score	Weighted score
Aesthetically Pleasing	2	3	6	9	18	2	4
Inexpensive	5	10	50	6	30	8	40
Easy to Assemble	3	8	24	10	30	9	27
Durable	4	2	8	9	36	10	40
Waterproof	1	2	2	9	9	10	10
Total			90		123		121

make the container easier and more intuitive to open and close for the sixth prototype. A scaled image of all 6 iterations can be seen F6 in Fig. 6.

Quote requests for injection molding the part were sent to several manufacturing companies. One company quoted $11,265 for the mold and $6.84 per part,[2] another quoted $6,470 for the mold,[3] and a third quoted $6,755 for mold and $2.04 per part.[4] While the variable cost was ideal, the fixed cost was over the laboratory's budget and considered prohibitive given the desire to minimize outreach program parts costs. A new manufacturing method was thus needed.

Further research revealed possibility of silicon molding and resin casting. At this point, discussions were made to outsource the silicone mold-making process to expert mold-makers. A request for a ballpark quote was sent to a manufacturing company (Thingsmiths LLC, 313 S State St #1, Ann Arbor, MI 48104). The mold itself was quoted at ~$500, while the part cost was around $25 for molding and $15 for 3D-printing. However, the time constraint would be an issue as each resin part took 15 min to fully cure. Mass-production would be slow and require many person-hours.

Code development

This project started with three main scripts to accomplish three tasks. Code was necessary to increase the voltage from the 9 V battery, to count the signals from the GM tube, and to display the data in an Android application. Originally, to increase the voltage from the 9 V battery to the necessary levels, an Arduino microcontroller was employed. The code for the microcontroller was from an online source who had made their code open to the public (Giametti 2018). The small sections of code needed to boost the voltage were used as an instructional reference and adapted to the needs of this project. Specifically, the Arduino was used solely for its pulse width modulation (PWM) capabilities to boost the voltage, something that will be explained in the results section of this report.

Everything having to do with this project on the Raspberry Pi was controlled through Python (Python Software Foundation, 9450 SW Gemini Drive, ECM# 90772, Beaverton, OR 97008). The Python script would count the signals from the GM tube over an adjustable period, set to 1 s by default. After investigating the various capabilities of both the Raspberry Pi and the Arduino, the Arduino was phased out, as everything this project was using the microcontroller for could be done by the Raspberry Pi as well. This meant that a small section of code controlling the PWM from a GPIO pin was added to the Python code (Crason 2013), and the Arduino AQ5 was removed from the design along with its code. After the decision to use Bluetooth to communicate between the Android device and the Raspberry Pi, a second Python script was written to acknowledge the Bluetooth connection, wait for a command from the app, and begin transmitting data once the command was received (CodingWithMitch 2018; Martin 2018).

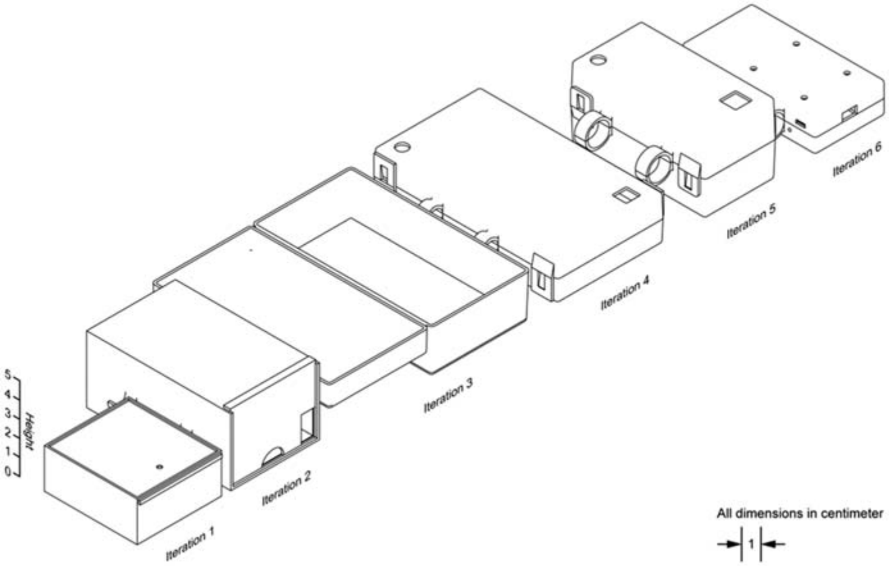


FIG. 6. Three-dimensional (3D) printed case design evolution, illustrating from left to right the first prototype detector container consisting of a simple box with sliding top and interior supports for circuit boards and power, the second prototype with probe support, internal shelf, and openings for cables, the third iteration designed with features essential to injection molding, a fourth self-closing design with two identical symmetrical parts to save on molding costs, a fifth modified design with reduced size for ergonomic purposes, and a final sixth case design with reduced thickness.

AQ4 [2] Personal communication, J. Clemons, Protolabs, 14 June 2018.

[3] Personal communication, D. Graham, XCentric Mold and Engineering, 15 June 2018.

[4] Personal communication, B. Bay, ICOMold, personal communication, June 14, 2018.

By far the longest section of code was written for the Android application meant to collect and display the data from the Raspberry Pi in a visual format, for the sake of creating heat maps. Everything was written in Java (JavaScript, Oracle Corporation, 500 Oracle Parkway, Redwood Shores, CA 94065) using Android Studio (Android Studio, Google LLC, 1600 Amphitheatre Parkway Mountain View, CA 94043). As this application needed to accomplish several objectives, the code was written in several sections before being combined.

The first section to be written covered the basic mapping capability. Google's pre-built Android software development kit (SDK) and application program interface (API) made up the bones of this section. Implementing the SDK and API meant that many necessary functions, such as location tracking for the Android device and marker creation on the map became built in and easily accessed. This first section was included in the application as a manual mapping option, where once the location tracking permission was granted, the user could manually input data to be dropped at their current device location using markers on a button push (CodingWithMitch 2018). These markers could then be converted into a heatmap, cleared from the map, or exported via email or text file.

The next several sections of code were meant to establish a Bluetooth connection between the Android device and the Raspberry Pi to transfer data between the two and automate the marking process. Multiple sections were necessary due to the complex nature of communication between two devices. The first step was to create an interface where users could select which device to connect to using Bluetooth. This meant that the application would need to work with the Android device to discover other devices, allow other devices to discover it, and to pair with other devices once they were found or had found it.

Once a Bluetooth connection could be successfully established, code was added to send a command to the Raspberry Pi to begin the data collection process and to switch to the map view. This required a location permission check, as with the manual mapping. At this point, the second Python script, mentioned before, was written, followed by another Java script to process the data received from the Raspberry Pi so that the mapping application could use it.

With the Bluetooth connection interface, command, and Python sections complete, everything was combined with a slightly altered version of the previously mentioned mapping section to create the automated mapping option of the application. With the basic functions of the application complete, buttons and additional functions were added to include a numeric data display and fully automated data collection on a user inputted time interval. As before, the options to create a heat map (Google Maps Platform 2018), export data, and clear all markers were also available after a marker was created.

With all the application capabilities written, an extended series of tests were done to work out all the bugs by downloading the application to an Android device and collecting data to see if anything went wrong. Using this method, a great deal of additional code was slowly added to ensure issues were not caused by the order that buttons were pressed. Once testing was confidently completed, the application and all code were uploaded to the internet. As large sections of the Java script were written using open source tutorials and guides, all the code involved with the project is open source and free to be downloaded, viewed, and edited at will.

As this is an educational outreach project, an explanation of the Python code will be given to participating students. It will be a good starting point for students, showing them how to import libraries, activate the GPIO pins to transmit and receive voltage signals, and demonstrate a few of the many capabilities of the Raspberry Pi. Due to the disproportionate complexity of the Android application, however, the Java code will not be explained to students, who are free to investigate the code should they feel the inclination on their own.

RESULTS

Final circuit design

The circuit designed for this project, shown in Fig. 2e, is comprised of three main parts. From pin 4, the 5-volt supply pin, and pin 6, the ground pin, begins a simple DC voltage booster that connects to the GM tube's cathode wires. The signals from the tube's anode wire go through a high pass signal filter and end with an inverter that converts the current signal to a voltage signal, before leading back to GPIO pins 31 and 32.

The voltage booster does exactly as the name implies: it boosts the voltage at a given point in the circuit. The voltage booster used in this project is a DC-DC converter, composed of transistor 1, the inductor, the diode, and capacitors 1 and 2. Specifically, it is a switched-mode power supply since it uses a diode, an inductor, and a transistor. This works by using the transistor as a switch, rapidly flipping it open and closed using the PWM function of the Raspberry Pi from pin 12. When the "switch" is closed, the inductor is charged with current, building an electric field while the capacitor provides current to the GM tube without affecting the rest of the circuit due to the diode. When the "switch" is opened, the charged inductor releases its charged current, acting like a second battery in series with the first, increasing the voltage and charging the capacitor. If the "switch" is flipped

Table 4. Do-it-yourself Geiger-Muller counter costs: Raspberry Pi and supporting components per unit built. Prices may vary for some components and listed prices are general estimates.

Raspberry Zero W				
Quantity	Item	Details	Supply: Manufacturer, Number, Description	Price
1	Raspberry Pi	Zero W	https://www.adafruit.com/product/3400	$10.00
1	Micro SD	8 GB	download Raspbian Stretch	$6.97
1	USB to micro USB cable	1 m or shorter	Stewart Connector, SC-2AMK003F, USB 2.0 Cable A Male to Micro B Male 0.914.m Shielded	$1.98
1	Mini HDMI to mini HDMI Cable	1 m or shorter	Variety Possible	$5.00
			Total Cost Per Setup:	$26.45

fast enough, the inductor will not fully discharge, resulting in a continuous elevated voltage (MIT 2018; Wikipedia 2018). In the context of this project, it was desired to boost the voltage across the GM tube since it requires an operating voltage of about 900 V while the Raspberry Pi only supplies 5 V.

After the GM tube voltage has been stepped up, the tube is able to create an output signal when it detects radiation. This output pulse takes the form of a very small current pulse in the circuit that is sent through a high-pass filter, composed of resistor 2, resistor 3, and capacitor 3, to reduce noise. Since the Raspberry Pi GPIO pins can only detect voltage fluctuations and not current signals, this filtered signal must be processed. The inverter, composed of transistor 2 and resistor 1, converts the current pulse output from the GM tube into a voltage pulse that can be read by the Raspberry Pi's GPIO pins (Circuit Fantasia 2018).

The final iteration is smaller than most of the early testing iterations and roughly half the size of the board in Fig. 3e due to the removal of the Arduino. Changing the Raspberry Pi model from the 3b to the Zero W further reduces the cost by about 22% of the original $90. The resulting final cost from all the changes and switching part vendors based on better deals is less than half the original $90 cost, sitting at about $40 per student. Upon completion, students will walk away with a working computer in the form of a Raspberry Pi, as well as a working detector with which they can explore and map radiological signals in their environment. More detailed descriptions of the costs are in Tables 3, 4, and 5 for the electronics, computer, and assembly stations. It is worth noting that prices of components will fluctuate depending on the source, so the $40 is not an exact value, but a higher-end estimate based on a simple search during 2018 of common university-approved suppliers in relatively small quantities.

Final case design

The final case design is still being discussed, as the multitude of options are still being explored. It was decided that manufacturing the cases in house would be too cost prohibitive, so more affordable alternatives are still being explored. Talks with various companies are still underway, and the more recent size change needs to be accounted for when deciding on a design. Among the designs researched is a clear, simple acrylic case, which can be purchased online for ~$2, offering the lowest price so far, but other more complex designs have their own appeal. The acrylic casing option reduces the cost of the device setup by about 3% and reduces build time. This portion of the design exercise proved very rich in terms of lessons concerning cost optimization of an actual product for a specific market size.

Final code scripts

All completed codes have been uploaded to GitHub (GitHub, 88

Table 3. Do-it-yourself Geiger-Muller counter costs: components to be assembled on the printed circuit board (PCB) for each unit built. Prices may vary for some components and listed prices are general estimates.

Internals				
Quantity	Item	Details	Our supply: manufacturer, number, description	Price
1	Custom PCB	Fits all parts in this list		$2.60
1	Geiger Muller Tube	Victoreen CDV700	Provided by David Miller of the University of Illinois	Free
1	Capacitor	330pF	VISHAY - F331K29Y5RN63J5R - CERAMIC CAPACITOR 330PF, 1000V, Y5R, 10%, RADIAL DISK	$0.08
2	Capacitor	10nF	VISHAY - S103M47Z5UN63J7R- D 1000V 10NF, 20% Z5U BULK E3	$0.11
1	High Speed Diode	UF4007	VISHAY - UF4007-E3/54 - FAST RECOVERY DIODE, 1A, 1KV, DO-204AL	$0.09
1	Inductor	15 mH	MURATA POWER SOLUTIONS - 17156C - INDUCTOR, FIXED IND 15mH 60MA 47 OHM	$1.09
1	NPN Transistor	STBV42-AP	STMICROELECTRONICS - STBV42-AP - TRANSISTOR, BIPOLAR, NPN, 400V, 1A, TO-92AP	$0.28
1	NPN Transistor	2N3904	ON SIMICONDUCTOR - 2N3904BU - BIPOLAR TRANSISTOR, NPN, 40V, TO-92	$0.06
3	Resistor	1.5kOhm	Stackpole Electronics Inc. - CF14JT1K50 - CARBON FILM RESISTOR, 1.5KOHM, 250mW, 5%	$0.10
1	Switch	Multipole		$0.82
			Total Cost Per Setup:	$5.54

Table 5. Do-it-yourself Geiger-Muller counter costs: Assembly station costs. Prices may vary for some components and listed prices are general estimates.

Assembly Station			
Quantity	Item	Details	Price
2	Wire	22AWG	$2.50
1	Soldering station	Varies	$40.00
2	Wire solder	82-117-ND multicore, 27AWG	$20.71
1	Monitor	Varies	$70.00
1	Keyboard and mouse	Varies	$22.00
		Total Cost Per Setup:	$155.21

Colin P Kelly Jr Street, San Francisco, CA 94107) free of charge, available for all to see, download, and edit. A small list of potentially helpful instructions is also included. To find it, one can go to Github.com, search DIY GM, and select the one uploaded by the first author of this paper. Of the completed codes, there are two final Python scripts shown in Fig. 7a and b, each working in almost the same way. Both use a loop and a timer to count signals from the GM tube and output the totaled results where signals would be received from the GPIO pin, and both use built in PWM functions to boost the voltage. This high output voltage can be changed by changing a single variable in the Python code, thus allowing for students to change to a different, more modern GM tube should they be interested. The key difference between the two scripts is that one of them contains sections of code meant to communicate over Bluetooth and does not begin counting until a command is received from the Android application.

As explained before, the Java code is much longer and more complex, split into many different sections based on their functions, called activities. It is worth noting that the layout and design of each activity in the application—what the user sees and interfaces with—has its own separate scripts. Each activity in the application calls its own layout script to determine what buttons, text boxes, menus, and so forth, exist, where they are placed, how much space they take up, and when they are visible vs. when they are not. This visibility can also be changed from the main activity and makes up a large section of the code to ensure none of the buttons and actions interfere with one another.

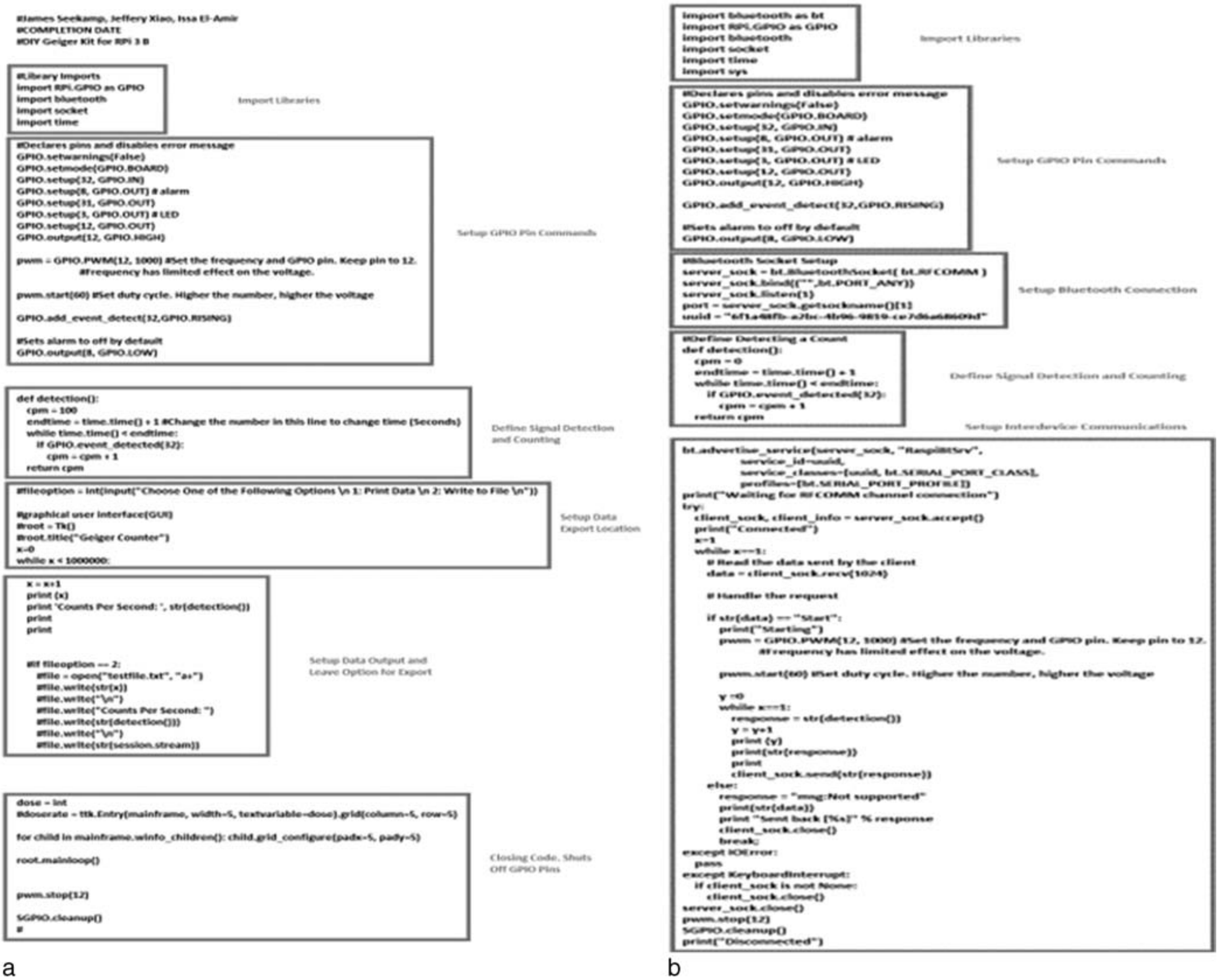
The first activity simply created a menu allowing for the user to select the manual mapping option or the automated mapping option. Should the manual mapping option be selected, the second activity begins, resulting in a permission check to see if the application could access the device location through Google Maps (Google Maps Platform 2018). Should it be found that permission was not granted, a permission request would appear that, if denied, would shut down the application.

Once the permission is granted, the map would be generated and would center on the device location. From here, the user may press the button labeled “Mark” to open a text box to enter numerical data into. Entering a number and pressing the “Set” button drops a marker on the current location with the latitude, longitude, and the input number. Once a marker is dropped, three more buttons appear: one that deletes all markers, one that generates the heat map, and one that opens a menu to export data, shown in Fig. 8a. There are no limits to the number of markers the user can drop, though device performance may change if too many data points are created. The rest of the code for this section is meant to prevent issues regarding the buttons, should they be pressed out of order. In total, this manual mapping activity requires three separate scripts: one main function that does all the necessary work, one setup script to establish the necessary permissions on the user’s device along with the Google Maps API, and a third for the layout of the visual display.

Automated data collection using Bluetooth required roughly twice as much code as the manual section. As explained before, the automated collection begins with a script that creates a menu from which a Bluetooth connection can be established. This menu is its own activity, requiring a section of code that handled checking if Bluetooth was available for the user’s Android device, detecting nearby Bluetooth devices, allowing other devices to detect the user’s device, and beginning the data mapping activity once a connection was made. A second script that handled creating the list of detected devices and a third script that established the Bluetooth communication were called by this first script.

Once a connection is made and the user begins the mapping activity, the functions are largely the same as the manual mapping with the key difference being the automated mapping receives data from the Raspberry Pi, not from the user’s input. An image of the view a user would see in this operating mode is shown in Fig. 8b. This data is handled by a separate script called by the automated mapping activity that also handled establishing the Bluetooth connection, used previously for the Bluetooth menu.

Assuming a stable connection and a working detector, a steady data stream will be displayed on the bottom of the screen. The user has two options for marking the map. The first option drops a marker based on data collected over 1 s. The second option is found at the bottom left corner of the screen and allows the user to set a time period in seconds. Once set, the application will total the number of counts from the GM tube over the set period and drop a marker once the time is up. This process



```
#James Seekamp, Jeffery Xiao, Issa El-Amir
#COMPLETION DATE
#DIY Geiger Kit for RPi 3 B

#Library Imports
import RPi.GPIO as GPIO
import bluetooth
import socket
import time

#Declares pins and disables error message
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(32, GPIO.IN)
GPIO.setup(8, GPIO.OUT) # alarm
GPIO.setup(31, GPIO.OUT)
GPIO.setup(3, GPIO.OUT) # LED
GPIO.setup(12, GPIO.OUT)
GPIO.output(12, GPIO.HIGH)

pwm = GPIO.PWM(12, 1000) #Set the frequency and GPIO pin. Keep pin to 12.
          #frequency has limited effect on the voltage.

pwm.start(60) #Set duty cycle. Higher the number, higher the voltage

GPIO.add_event_detect(32,GPIO.RISING)

#Sets alarm to off by default
GPIO.output(8, GPIO.LOW)

def detection():
  cpm = 100
  endtime = time.time() + 1 #Change the number in this line to change time (Seconds)
  while time.time() < endtime:
    if GPIO.event_detected(32):
      cpm = cpm + 1
  return cpm

#fileoption = int(input("Choose One of the Following Options \n 1: Print Data \n 2: Write to File \n"))

#graphical user interface(GUI)
#root = Tk()
#root.title("Geiger Counter")
x=0
while x < 1000000:
  x = x+1
  print (x)
  print 'Counts Per Second: ', str(detection())
  print
  print


  #if fileoption == 2:
    #file = open("testfile.txt", "a+")
    #file.write(str(x))
    #file.write("\n")
    #file.write("Counts Per Second: ")
    #file.write(str(detection()))
    #file.write("\n")
    #file.write(str(session.stream))


dose = int
#doserate = ttk.Entry(mainframe, width=5, textvariable=dose).grid(column=5, row=5)

for child in mainframe.winfo_children(): child.grid_configure(padx=5, pady=5)

root.mainloop()


pwm.stop(12)

SGPIO.cleanup()
#
```

```
import bluetooth as bt
import RPi.GPIO as GPIO
import bluetooth
import socket
import time
import sys

#Declares pins and disables error message
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(32, GPIO.IN)
GPIO.setup(8, GPIO.OUT) # alarm
GPIO.setup(31, GPIO.OUT)
GPIO.setup(3, GPIO.OUT) # LED
GPIO.setup(12, GPIO.OUT)
GPIO.output(12, GPIO.HIGH)

GPIO.add_event_detect(32,GPIO.RISING)

#Sets alarm to off by default
GPIO.output(8, GPIO.LOW)

#Bluetooth Socket Setup
server_sock = bt.BluetoothSocket( bt.RFCOMM )
server_sock.bind(("",bt.PORT_ANY))
server_sock.listen(1)
port = server_sock.getsockname()[1]
uuid = "6f1a48fb-a2bc-4b96-9819-ce7d6a68609d"

#Define Detecting a Count
def detection():
  cpm = 0
  endtime = time.time() + 1
  while time.time() < endtime:
    if GPIO.event_detected(32):
      cpm = cpm + 1
  return cpm

bt.advertise_service(server_sock, "RaspiBtSrv",
          service_id=uuid,
          service_classes=[uuid, bt.SERIAL_PORT_CLASS],
          profiles=[bt.SERIAL_PORT_PROFILE])
print("Waiting for RFCOMM channel connection")
try:
  client_sock, client_info = server_sock.accept()
  print("Connected")
  x=1
  while x==1:
    # Read the data sent by the client
    data = client_sock.recv(1024)

    # Handle the request

    if str(data) == "Start":
      print("Starting")
      pwm = GPIO.PWM(12, 1000) #Set the frequency and GPIO pin. Keep pin to 12.
          #frequency has limited effect on the voltage.

      pwm.start(60) #Set duty cycle. Higher the number, higher the voltage

      y =0
      while x==1:
        response = str(detection())
        y = y+1
        print (y)
        print(str(response))
        print
        client_sock.send(str(response))
    else:
      response = "msg:Not supported"
      print(str(data))
      print "Sent back [%s]" % response
      client_sock.close()
      break;
except IOError:
  pass
except KeyboardInterrupt:
  if client_sock is not None:
    client_sock.close()
server_sock.close()
pwm.stop(12)
SGPIO.cleanup()
print("Disconnected")
```

AQ6 **FIG. 7.** Python code for smart detector operation, detailed and documented for a) simple manual input of readings and their location where they were recorded, and b) automated Bluetooth capture of radiation readings and location.

repeats until manually canceled or the application is closed. Lastly, as with the manual option, once a marker is placed, the options to delete all markers create a heatmap shown in Fig. 8c, and export the data appear.

Comparison to other devices

Simple GM detectors and other do-it-yourself detectors have been available on the market for some time (Jeff Keyzer, MightyOhm, see also John Giametti, DIY Geiger Counter, and Images SI Inc., 109 Woods of Arden Road, Staten Island, NY 10312). This kind of project is not particularly groundbreaking. The difference, and advantage of this project and design over others, is the emphasis on and the scope of learning, the completion of all supplied parts, and its low cost. The most obvious and important difference is the incorporation of the Raspberry Pi into this design. From the Raspberry Pi, students can learn computer programing through Python, broadening the scope of what participating students will be exposed to, with the trade-off of having a slightly larger device. Along with this, other boards will almost always cost more than this design, some costing upwards of $100 compared to the $40 seen here. This design is preferable when compared with models of similar price, as some will not come with crucial parts, such as GM tubes due to their cost. While the model 6993 tubes included here are far from professional, they complete their purpose, are very affordable, and students are free to upgrade their tubes at their own discretion by following the instructions given to them.

Student exercises

A short set of exercises was developed for students to try for the purpose of testing their devices once they had completed the assembly of their GM detectors, similar to the device shown in Fig. 9 F9

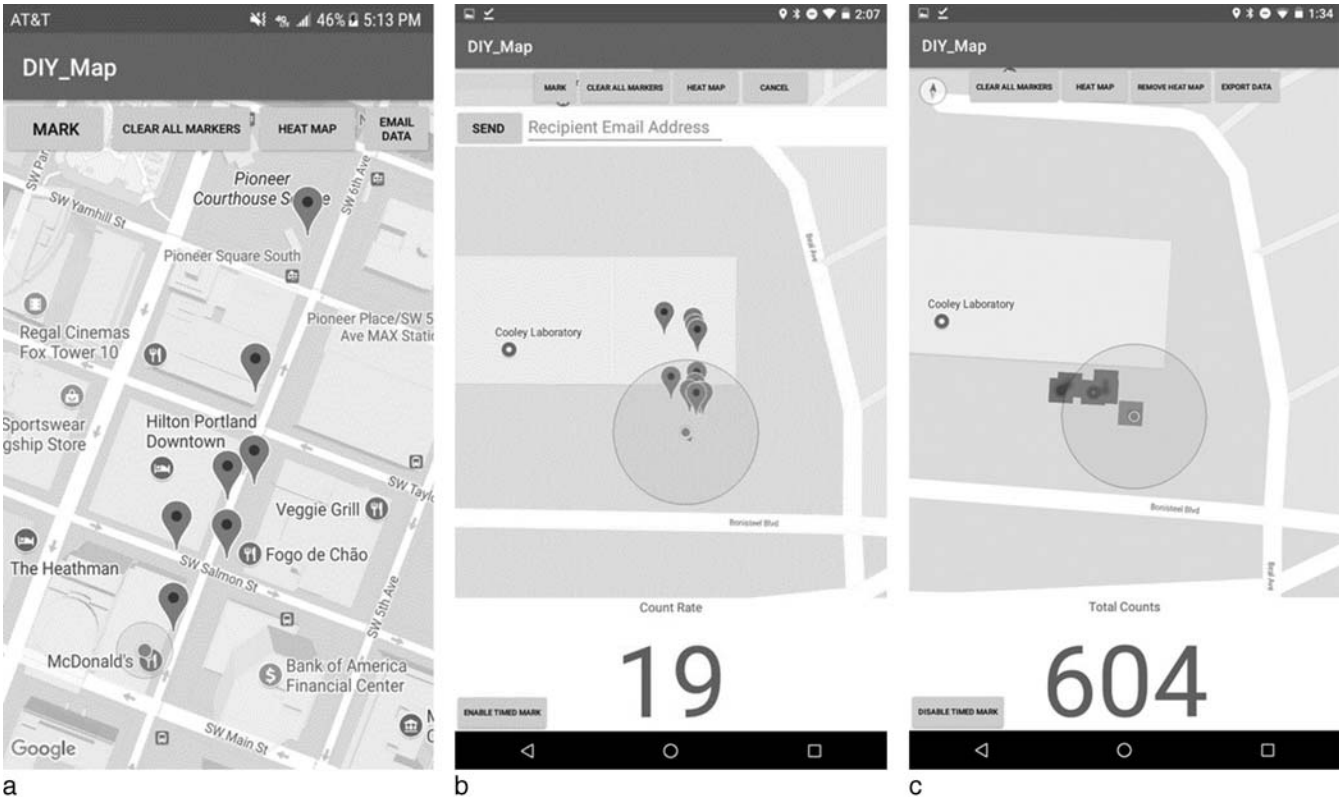


FIG. 8. Smartphone display of mapped radiation, showing a) simple manual input of readings and location information; b) automated data acquisition using Bluetooth; and c) heatmap of data collected using automated data acquisition.

without any final case design but a protective cover over a Raspberry Pi Zero W. Students may use uranium-glazed ceramics, such as Fiestaware (The Homer Laughlin China Company, 672 Fiesta Drive, Newell, WV 26050) or any other relatively weak, but reliable radiation source, for these activities. They can make measurements at varying distances from the source and use common notebooks as shielding to explore the effects of distance and attenuation. Additionally, collection times can be changed to demonstrate the effects on counting statistics.

CONCLUSION

Through the assembly of a working GM detector, students will be exposed to and learn more about several valuable STEM topics, including but not limited to radiation detection, circuit building, soldering, and computer programming, with the long-term objective of drawing more people into STEM related fields and careers. Creating an affordable, easily understood GM detector that used a user-friendly interface in the form of a Raspberry Pi and Android application took several iterations due to learning of fundamentals by the design team members, improvements realized through experience, and modifications to ultimately reduce cost. The result of roughly 800 person-hours and $1,000 in

FIG. 9. Final assembled design, shown attached to a Raspberry Pi Zero W in a transparent case.

development costs was an outreach project that should take inexperienced students ~2 h to build and about 30 min to test. Students will be able to keep their finished detectors, allowing interested individuals to take the initiative and learn more on their own about electronics, radiation detector design, or simply radiation found in their environment.

Acknowledgments—GM tubes salvaged for this project were provided by David Miller, University of Illinois, with the approval of the Health Physics Society. Thanks are also extended to Robert M. Golduber, John R. Kuchta, Emily H. Kwapis, Weronika Myslak, and Kevin Mingyuan Bi, all University of Michigan undergraduate students at the time this work was performed, for testing and debugging help. This work was funded by the Consortium for Verification Technology under the Department of Energy National Nuclear Security Administration, award number DE-NA0002534.

REFERENCES

AQ7 Altium (CircuitStudio). Understanding the nuances between breadboard projects and prototype layouts [online]. 2018. Available at ***https://resources.altium.com/pcb-design-blog/understanding-the-nuances-between-breadboard-projects-and-prototype-layouts***. Accessed 22 August 2018.

Apple Inc. iPhone X – Technical Specifications – Apple [online]. 2018. Available at ***https://www.apple.com/iphone-x/specs/***. Accessed 22 August 2018.

Bergen RJ, Harvey JA, Kearfott KJ. Performance of vintage direct reading pocket ionization chambers. Health Phys 98(Suppl 2): S56–S62; 2010. DOI 10.1097/HP.0b013e3181cd3e18.

Carlson LE, Sullivan JF. Hands-on engineering: learning by doing in the integrated teaching and learning program. Internat J Engineering Ed 15:20–31; 1999.

Center for Nuclear Science and Technology Information (American Nuclear Society). Event June 16: detecting radiation in our radioactive world [online]. 2018. Available at ***www.nuclearconnect.org/about/events***. Accessed 22 August 2018.

Circuit Fantasia. Inventing circuits on the whiteboard: passive current-to-voltage converter [online]. 2018. Available at ***www.circuit-fantasia.com/circuit_stories/inventing_circuits/i-to-v_converter/i-to-v_converter.htm***. Accessed 14 July 2018.

CodingWithMitch. Home [online]. 2018. Available at ***https://www.youtube.com/channel/UCoNZZLhPuuRteu02rh7bzsw***. Accessed 22 August 2018.

Corter JE, Esche SK, Chassapis C, Ma J, Nickerson JV. Process and learning outcomes from remotely-operated, simulated, and hands-on student laboratories. Computers Ed 57:2054–2067; 2011.

Croston B. "Raspberry-Gpio-Python" SourceForge [online]. 2018. Available at ***sourceforge.net/p/raspberry-gpio-python/wiki/PWM/***. Accessed 23 April 2018.

Csik BJ, Laue HJ, Raisic N. Manpower development for nuclear power programmes. IAEA Bulletin 21: 65–69; 1979.

Deutsch RW, Whitney JW. Nuclear manpower crisis ahead. Columbia, MD: General Physics Corporation; 1974.

Felder RM, Silverman LK. Learning and teaching styles in engineering education. Engineering Ed 78:674–681; 1988.

Folger JK, Meeks ML. Educated manpower: Key to nuclear development. In: Redding S, ed. Nuclear energy in the south. Baton Rouge, LA: Louisiana University Press; 1957.

Giametti J. DIYGeigerCounter [online]. 2018. Available at ***https://sites.google.com/site/diygeigercounter***. Accessed 22 August 2018.

Gibson HL, Chase C. Longitudinal impact of an inquiry-based science program on middle school students' attitudes toward science. Sci Ed 86:693–705; 2002.

Google Maps Platform. Google maps android heatmap utility [online]. 2018. Available at ***https://developers.google.com/maps/documentation/android-sdk/utility/heatmap***. Accessed 6 May 2018.

Haik Y. Engineering design process. Pacific Grove, CA: Brooks/Cole-Thompson Learning; 2003.

Hansen T. Meeting the power industry's workforce challenges. Power Engineering 112:40–43; 2008.

Health Physics Society. Resources for your classroom [online]. 2018. Available at ***http://hps.org/sciencesupport/classroomresources.html***. Accessed 4 September 2018.

Holtzapple MT, Reece WD. Foundations of engineering. New York: McGrawHill- Education; 2002.

Hulleman CS, Harackiewicz JM. Promoting interest and performance in high school science classes. Sci 326:1410–1412; 2009. DOI 10.1126/science.1177067.

Johnson RC. Manpower requirements in the nuclear power industry, 1982-1991. Oak Ridge, TN: Oak Ridge Associated Universities, Inc.; ORAU-205; 1982. DOI 10.2172/6758630.

Kaiser D. Cold War requisitions, scientific manpower, and the production of American physicists after World War II. Hist Stud Phys Biol Sci 33:131–159; 2002. DOI 10.1525/hsps.2002.33.1.131.

Klahr D, Triona LM, Williams C. Hands on what? The relative effectiveness of physical versus virtual materials in an engineering design project by middle school children. J Res Sci Teach 44:183–203; 2007. DOI 10.1002/tea.20152.

Knoll GF. Radiation detection and measurement. Hoboken, NJ: John Wiley and Sons; 2010.

Kuhn JW. Scientific and managerial manpower in nuclear industry. New York: Columbia University Press; 1966.

Powers P. The changing manpower picture. Scientific Monthly 70: 165–171; 1950.

Markowitz DG. Evaluation of the long-term impact of a university high school summer science program on students' interest and perceived abilities in science. J Sci Ed Technol 13:395–407; 2004.

Martin T. How to setup Bluetooth on a Raspberry Pi 3 [online]. 2018. Available at ***https://www.cnet.com/how-to/how-to-setup-bluetooth-on-a-raspberry-pi-3/***. Accessed 6 May 2018.

Massachusetts Institute of Technology. Geiger-Müller counter circuit theory [online]. 2018. Available at ***https://ocw.mit.edu/courses/nuclear-engineering/22-s902-do-it-yourself-diy-geiger-counters-january-iap-2015/readings/MIT22_S902IAP15_geigr_ckt.pdf***. Accessed 12 June 2018.

McDonald JT, West WG, Kearfott KJ. An evaluation of the Kearney fallout meter (KFM) radiation detector constructed from commonly available household materials. Health Phys 87:S52–S57; 2004.

Mestre JP. Implications of research on learning for the education of prospective science and physics teachers. Phys Ed 36:44; 2001.

Moeller DW. Meeting radiological health manpower needs. Am J Public Health 61:1938–1946; 1971.

Powers P. The changing manpower picture. Scientific Monthly 70: 165–171; 1950.

Roehl A, Reddy SL, Shannon GJ. The flipped classroom: An opportunity to engage millennial students through active learning strategies. J Family Consumer Sci 105: 44–49; 2013.

Shilling CW. Implications of nuclear science developments for graduate training. Scientific Manpower 8:25; 1960.

Singer S, Smith KA. Discipline-based education research: understanding and improving learning in undergraduate science and engineering. J Engineering Ed 102: 468–471; 2013. DOI 10.1002/jee.20030.

Stevenson W. Personnel requirements, education, and training for civilian nuclear activities, 1984-2000. Oak Ridge, TN: Oak Ridge Associated Universities, Inc.; ORAU-231; 1984.

Stohr-Hunt PM. An analysis of frequency of hands-on experience and science achievement. J Res Sci Teach 33:101–109; 1996. DOI 10.1002/(SICI)1098-2736(199601)33:1<101:AID-TEA6>3.0.CO;2-Z.

Vernier Software and Technology. Available at ***https://www.vernier.com/***. Accessed 4 September 2018.

Was GS, Martin WR. Manpower supply and demand in the nuclear industry. La Grange Park, IL: American Nuclear Society, Nuclear Engineering Department Heads Organization (NEDHO); 2000.

Wellington J. Formal and informal learning in science: The role of the interactive science centres. Phys Ed 25:247; 1990.

Wikipedia. Boost converter [online]. Available at ***https://en.wikipedia.org/wiki/Boost_converter***. Accessed 22 August 2018.

Williams E. Use a cheap pin diode as a Geiger counter [online]. 2014. Available at: ***https://hackaday.com/2014/10/25/use-a-cheap-pin-diode-as-a-geiger-counter/***. Accessed 1 July 2018.

Wilson ME. Teaching, learning, and millennial students. New Directions Student Services 2004: 59–71; 2004. DOI 10.1002/ss.125.

Wogman NA, Bond LJ, Waltar AE, Leber RE. The nuclear education and staffing challenge: Rebuilding critical skills in nuclear science and technology. J Radioanalyt Nucl Chem 263:137–143; 2005. DOI 10.1007/s10967-005-0027-z.

Xue Y, Larson RC. STEM crisis or STEM surplus? Yes and yes: Monthly Labor Review, Bureau of Labor Statistics, U.S. Department of Labor [online]. 2015. Available at ***https://www.bls.gov/opub/mlr/2015/article/stem-crisis-or-stem-surplus-yes-and-yes.htm***. Accessed 22 August 2018.

AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES

AQ1 = Please check if authors name are correctly captured for given names (in red) and surnames (in blue) for indexing after publication.

AQ2 = Add complete citation for Johnson 1991 to reference list.

AQ3 = Williams 2018 changed to 2014 to match reference list. OK?

AQ4 = Add contact addresses for personal communication citations.

AQ5 = Add complete citation for Crason 2013 to reference list.

AQ6 = Provide high resolution images for figures 7a and 7b.

AQ7 = Add page range to Folger and Meeks 1957.

END OF AUTHOR QUERIES