Introducing the Internet-of-Things to the Next Generation of Engineers

Dr. Samuel J. Dickerson, University of Pittsburgh

Dr. Samuel Dickerson is an assistant professor at the University of Pittsburgh Swanson School of Engineering. His general research interests lie in the area of electronics, circuits and embedded systems and in particular, technologies in those areas that have biomedical applications. He has expertise in the design and simulation of mixed-signal integrated circuits and systems that incorporate the use of both digital and analog electronics, as well as optics, microfluidics and devices that interface to the biological world. Prior to joining the University of Pittsburgh faculty he was a co-founder and the president of Nanophoretics LLC, where he led the research and development of a novel dielectrophoresis-based lab-on-chip technology for rapidly detecting drug-resistant bacteria strains. Dr. Dickerson is also interested in enhancing undergraduate engineering education, and investigates new and innovative methods for improving the learning experience for electrical and computer engineering students.

Introducing the Internet-of-Things to the Next Generation of Engineers

Abstract

The world is currently on the verge of the next industrial revolution, the Internet-of-Things, the movement towards embedding everyday objects with intelligence and the ability to wirelessly communicate information to the cloud. The Internet-of-Things promises to drastically change several aspects of our lives, from the way business is conducted to how we go about otherwise routine day to day activities. This oncoming revolution will create a workforce need that those in STEM education fields must work now to fill via efforts to broaden participation in electrical and computer engineering. In particular, this specialized future workforce will have to be bolstered through K-12 outreach and recruitment of students from traditionally underrepresented groups in engineering.

In this paper, we present our University's efforts to contribute to this need by way of a hands-on activity designed for high school students. The workshop was devised to achieve three primary goals: 1) Encourage consideration of a career in electrical and computer engineering 2) Build excitement about the Internet-of-Things and provide students with a future technical focus and 3) Introduce students to the fundamental building blocks that make up the Internet-of-Things. During this activity, students complete a project in which they first construct a circuit to read data from a temperature sensor using a microcontroller platform. The students then write software to transmit that data over a short-range wireless network and then eventually to an Internet-connected device. Finally, that information is used to actuate a DC motor, thereby emulating the same loop carried out by many Internet-of-Things devices, creating data that originates from a sensor, goes to the cloud and then comes back to perform action.

We report on two offerings of this workshop and present results in various forms. In addition to student surveys, an observation protocol was used to collect information on the level of activity and student engagement. Finally, samples of student work were directly assessed in terms of their quality and completeness. The results show that the activity is engaging and is successful in meeting its three stated goals. We also provide lessons learned, suggestions for educators who wish to deploy similar activities and propose ideas to improve future offerings.

1. Introduction

The Internet-of-Things (IoT) refers to the idea of embedding objects with intelligence, the ability to sense information about its environment and network connectivity. These intelligent, interconnected "things" currently range from smart appliances and wearable devices to machines and vehicles. The list of network connected "things" is growing rapidly, leading to some estimates that in the near future there will be billions and billions of network connected "devices" (Gartner, 2015). The Internet-of-Things has changed the world that we live in and is continuing to evolve rapidly.

The Internet-of-Things provides a unique opportunity to encourage and broaden participation in Electrical and Computer engineering. While the Internet-of-Things has applications in many diverse areas and involves several engineering disciplines, Electrical and Computer engineering technologies and methods are its core (i.e. networking, embedded computing, sensing and data analysis, Figure 1). Furthermore, IoT is something that everyone is currently seeing proliferate before their very own eyes, IoT technologies are tangible and their impact is highly visible. As a result, students are excited to learn more about IoT and eventually pursue employment in IoT-related fields.

Along with the excitement brought about by IoT comes a significant educational challenge, particularly if using IoT as a form of outreach. Even within the domain of Electrical and Computer Engineering, the skillset needed in order to design an IoT device is quite broad. If one is to develop a true "IoT" workshop for students, care has to be taken to differentiate the field from other closely related areas, such as embedded computing. What separates IoT from embedded systems is the combination of sensing, embedded computing, networking and the cloud altogether to perform a dedicated task. Furthermore, in order to build excitement among students in a workshop setting, the associated activities must be short, but meaningful, and leave the students with a feeling of accomplishment.

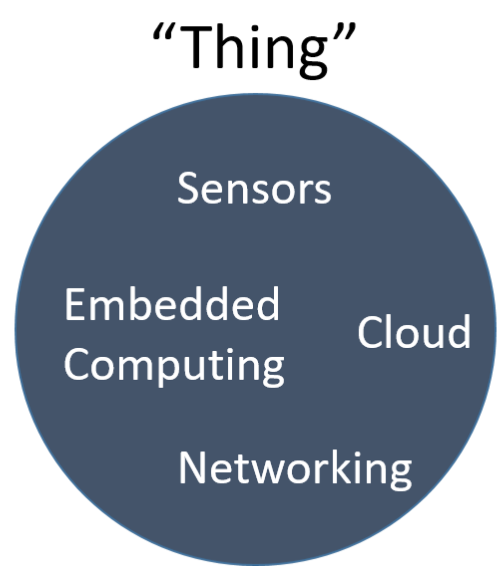

Figure 1. Primary components of an Internet-of-Things technologies

In this paper, a new Internet-of-Things workshop designed for engineering outreach is presented. The workshop meets its fundamental goals by incorporating all aspects of IoT in a short, one-day format. The rest of the paper is organized as follows: In the next section related work is briefly discussed, after which a detailed discussion of the workshop structure and activities is given. Results from two separate offerings of the workshop are provided in the form of direct and indirect assessments. Conclusions are then provided along with suggestions for future offerings.

2. Related Work

There are numerous reports of pre-college workshops designed to broaden participation in engineering where the activities center on microcontrollers, sensors and the like. A large portion of those activities involve robotics. However, there are no reports of outreach activities that are portrayed as being directly related to the Internet-of-Things and very few that are similar in nature. One recent example that does share some similarities to the work presented here is billed as a wireless sensor network activity (Feaster, 2015). The activity introduces some similar concepts such as the combination sensors, embedded computing and wireless networking but does not involve cloud computing services. Moreover, the activity was designed for middle school students and participants are not exposed to the same level of depth presented here. What is presented in this paper is a truly novel workshop and as of the time of this publication is the first of its kind, incorporating all facets of the Internet-of-Things.

3. Instructional Methods

The workshop was devised to achieve three primary goals: 1) Encourage consideration of a career in electrical and computer engineering 2) Build excitement about the Internet-of-Things and provide students with a future technical direction and 3) Introduce students to the fundamental building blocks that make up the Internet-of-Things. For this workshop, students are given a series of IoT design challenges put in the context of a home automation system. In this section, the organization and content of the workshop are presented in detail.

3.1 Internet-of-Things Hardware Platform

The microcontroller platform is the centerpiece of this hands-on activity. Since the theme of the workshop is the Internet-of-Things, it is required that the platform also incorporate network connectivity. Additionally, since the target audience is high-school students, it is desirable that the device is easy for students to use.

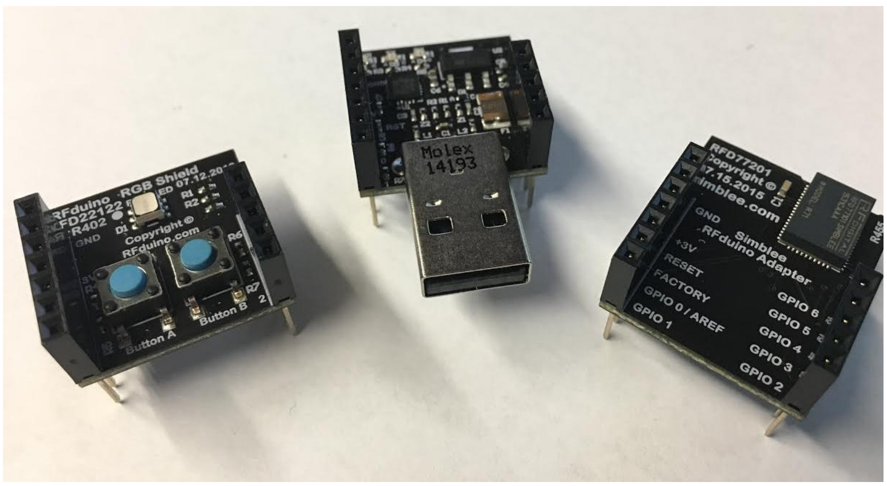

Figure 2. Simblee microcontroller platform used for IoT workshop.

The Simblee development kit was found to satisfy the educational hardware platform requirements (Simblee). The development kit is comprised of three components (figure 2), a microcontroller with an integrated Bluetooth module, a USB programmer, easily accessible General purpose I/O as well as integrated LEDs and buttons. In addition to these features, the platform provides support for cloud-based tools and mobile application development. The usage of this device and the accompanying tools is discussed further in the following sections.

3.2 Curriculum Design

As of the time of this publication, this IoT workshop has been offered twice. After the first offering, the program was reevaluated and improvements were made. In this section, the curriculum for the improved offering is discussed. Results from both offerings and discussions of improvements made are included in the results.

The target audience for this workshop are high school students. While some students may have previous programming or electronics experience, the activity is designed under the assumption that the students have none, and that this will be their first exposure to computer programming and electronic prototyping. The activity is designed to be a one-time workshop of approximately

3 hours in duration. Students are asked to work together in pairs and the activity is structured as a semi-self-directed exercise. The students are given packets that contain lecture notes and instructions on the day's tasks. Periodically throughout the lecture, at planned intervals, the class is brought to attention and brief lectures are given setting up the next portion of the activity and providing instruction on how to complete the current task. This approach was selected because of the various student backgrounds, some have experience working with such technologies and some do not, therefore this presentation approach allows for students to work at their own pace and not get bored waiting on others or become frustrated with too fast of a pace. This is critically important as the goal is to excite students about engineering, and that can be achieved regardless of whether or not the activity is completed in its entirety. This also requires a higher instructor to student ratio so that help is available when needed. For the offerings being reported on here, workshop was led by a University faculty member, with undergraduate and graduate students used for support.

3.2.1 Workshop Introduction

The workshop begins with an overview of the Internet-of-Things. Time is spent on defining what is meant by IoT, the impact of IoT today's world, its expected future impact and then specific examples that the students can relate to are given. Regardless of the application area (biomedical, power/energy, home automation, etc.) the commonalities of IoT devices in these areas are emphasized:

- ***Sensors***
- ***Embedded Computing***
- ***Wireless Networking***
- ***Cloud Computing / Analytics***

One of the main takeaways that it is hoped that students learn from the activity is what makes IoT something new, different and exciting is that the IoT is the combination of these four areas to perform some sort of dedicated task.

In order to provide IoT-centric student tasks, the students were given a series of design challenges, all pertaining to home automation and were told that they were tasked with designing various parts of an internet-connected smart home. In order to give students a brief glimpse into the engineering design process, the students were first given non-technical, high-level customer requirements (figure 3):

<u>Customer Requirements</u>

The customer wants an IoT device that provides the ability to:

1. Control home lights from a remote location
2. Remotely monitor ambient temperature of rooms in a home
3. Automatically adjust cooling system based on current weather

Next, time was spent, as a group, translating those customer requirements into technical specifications. In this portion of the workshop, the microcontroller and its capabilities are introduced. It is explained how the microcontroller is the "brains" behind the proposed IoT solution and acts as a way for us to control each hardware subsystem via software. The students are then helped to appreciate that in order to satisfy the customer's requirements, the electrical and computer engineering teams (the students) must devise ways to:

Figure 3. Customer Requirements

- Turn a light (LED) on/off via an electrical signal
- Read in temperature data via a sensor
- Based on the temperature values read, electronically activate a fan motor
- Wirelessly transmit temperature data, the state of the light and the fan via a Bluetooth Network to a hand held device
- Wirelessly transmit user input, via Bluetooth, indicating user control of light switches to the microcontroller
- Allow the hand held device to push data to and receive data from an 802.11 wireless network for further processing
- Enable the user to observe and control the system via a easy to use Graphical User Interface that abstracts away the technical details from the customer

Although the workshop is brief, students work to satisfy of all of the above technical requirements via to varying degrees. Spending sufficient time on this introductory context setting portion the activity is important, as it transforms the subsequent tasks from being a series of abstract technical exercises (e.g. reading data from sensors) to design challenges that have motivating purposes.

3.2.2 Introduction to Schematics, Electronic Prototyping and Sensor Interfacing

The first task given to students is construct the temperature sensing circuitry. Prior to starting this activity, students are given a short primer on digital systems and the concept of representing information and numbers with binary encoded numbers. It is also explained to students how with digital electronics, engineers use logic abstractions and individual bits are thus represented as either high or low voltages.

To further that concept, students are then given datasheets for the temperature sensor they eventually use, the TMP102 (Texas Instruments). The TMP102 is a digital temperature sensor with a two wire serial interface. Figure 4 shows a picture of the sensor given to students. First, students are taken to the datasheet and shown the binary code mapping used by the sensor to represent detected temperature values (Table 1).

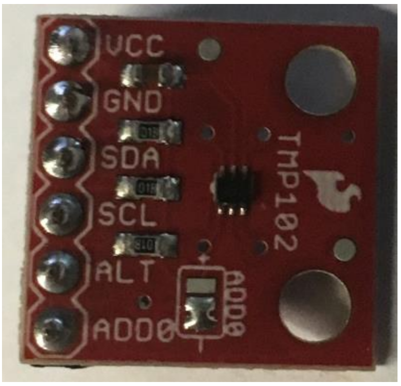

Figure 4. Digital Temperature Sensor

Table 1. Temperature to Binary Code Mapping

Temperature (°C)	Binary Code
100	0 0110 0100 0000
75	0 0100 1011 0000
50	0 0011 0010 0000
25	0 0001 1001 0000
0.25	0 0000 0000 0100
0	0 0000 0000 0000
-0.25	1 1111 1111 1100
-25	1 1110 0111 0000

After examining the datasheet, the concept of Machine-to-Machine communication is introduced, the idea that electronic devices must use a specified protocol in order to be able to communicate without human intervention. Machine-to-Machine communication is one of the core abilities integrated in almost all IoT devices. In the context of the student exercise, a mechanism is needed in order to transmit temperature data to the microcontroller. The machine-to-machine protocol used to communicate with the sensor is I^2C (NXP). The I^2C protocol is not

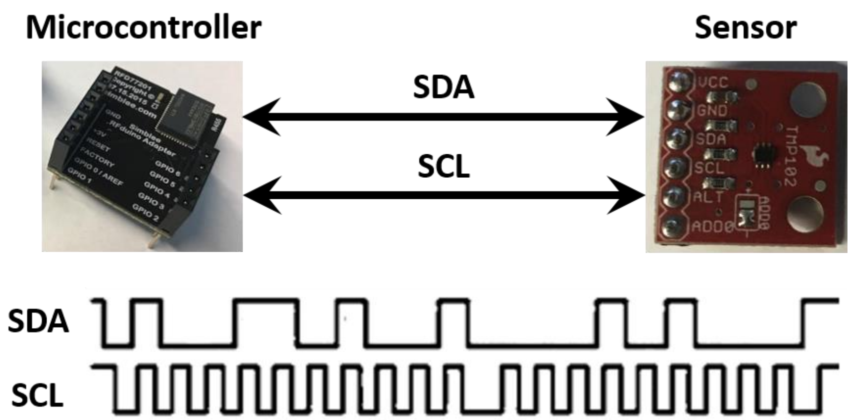

Figure 5. Temperature sensor to microcontroller I^2C Machine-to-Machine communication

explained in its entirety to the students, but rather a high-level description is given and a small example of how a temperature value would be transmitted from the sensor to the microcontroller is provided (figure 5).

After being given the preceding information, the students begin the task of constructing the prototype. As no prior experience is assumed, students are given a brief demonstration on how to connect devices to one another via an electronic prototyping breadboard. The concept of using schematics to represent circuit connectivity is explained, some standard electrical symbols are explained (e.g. power, ground, etc.) and a small example is given. After that explanation, students are told the function each of the pin on the temperature sensor, the corresponding pins on the microcontroller, and asked to complete a schematic. Symbols for components (i.e. power, ground and the temperature sensor) are given to the student. Figure 6 provides an example of a completed student schematic. After completing the schematic, students are prepared and ready to write software for the microcontroller.

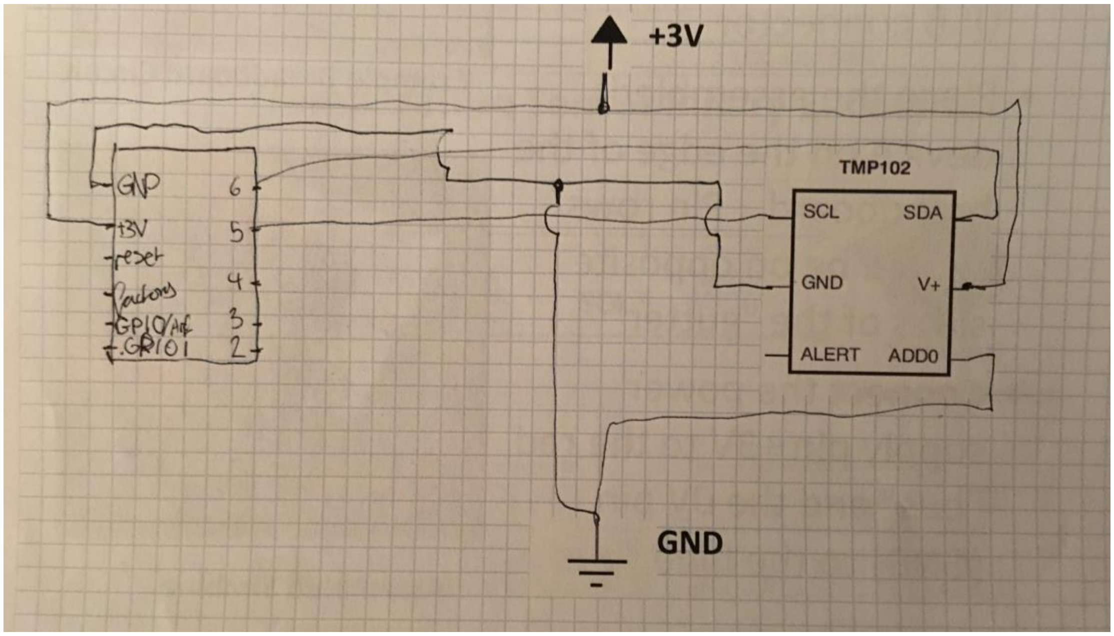

Figure 6. Sample student schematic showing sensor and microcontroller connectivity

3.2.3 Introduction to Software Development for Embedded Computing

The purpose of the subsequent phase of the workshop is to teach students the fundamentals of software development in the context of an embedded system. No prior computer programming experience was required or assumed for this workshop. It is explained to students that with embedded systems, one has the ability to control hardware (e.g. sensors, lights, motors, etc.) via software. The hardware development platform used is not an Arduino, but can be made to use the simple Arduino programming environment. The basics of programming with the Arduino IDE is briefly explained (figure 7). For the first programming task is for the students to display their group name and the pin numbers associated with the sensor interface and the I^2C address of the sensor to the debug terminal.

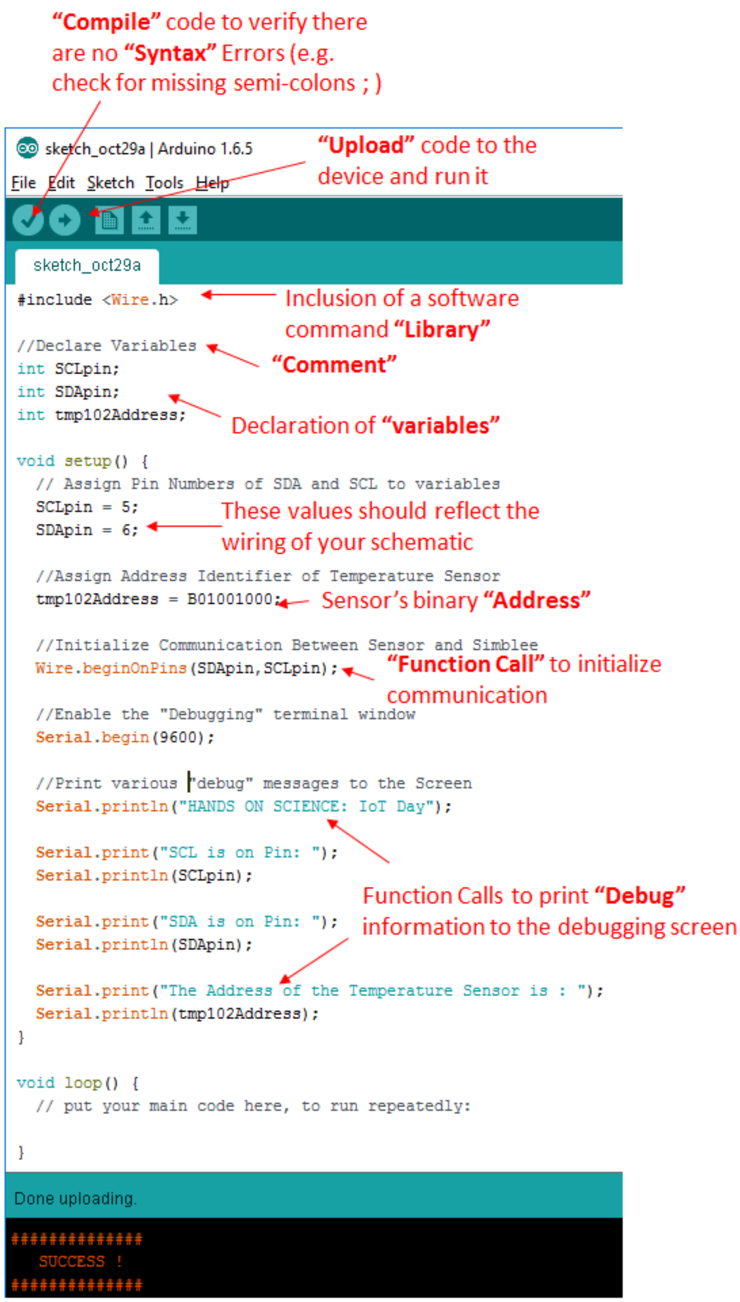

Figure 7. Student's first introduction to the microcontroller development environment and embedded programming fundamentals

Students are then shown how to make use of subroutines in order to carry out tasks that will be carried out repeatedly. In order to conserve time, students are provided with a subroutine that returns the sensed temperature in Celsius (figure 8). Some of the lower-level bit manipulations that are required in order to read data from the sensor is discussed, but not emphasized, as gaining a deep understanding of that aspect is beyond the planned scope of the activity. As an activity, students are asked to write a new subroutine that returns the temperature in Fahrenheit as opposed to Celcius. This task requires minimal modifications to the provided function, but

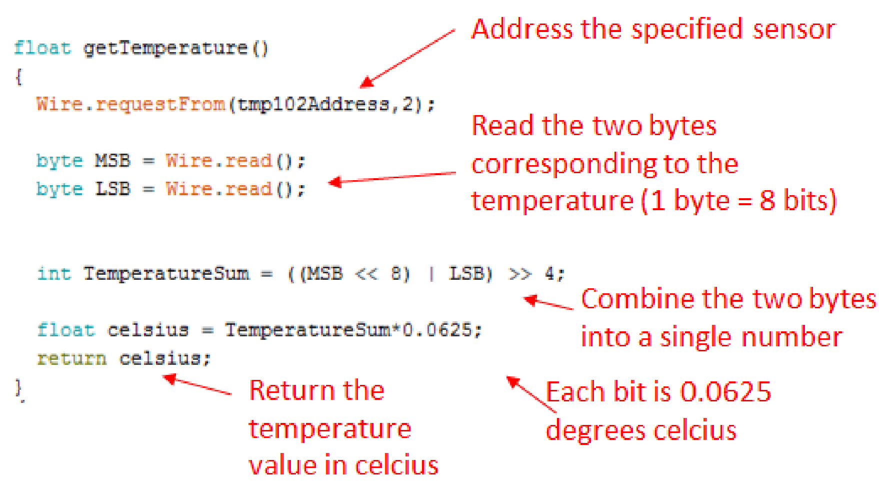

Figure 8. Subroutine for reading temperature data that students are asked to modify

gives the student the opportunity to gain experience using and modifying subroutines.

After students have gained experience in using the programming environment, using variables and function calls, the IF/ELSE construct is introduced. This leads into a discussion of algorithms and the importance of designing algorithms during the embedded software development process. The next major task is for students to interface an LED to the microcontroller and then cause the light to turn on for a duration of 1 second if the sensed temperature is greater than a specified temperature threshold (76º F). Students are provided with nothing more than an LED (with a description of the role of the anode and cathode terminals) and a set of subroutines for initializing input/output pins, writing digital outputs to those pins and delaying the execution of software for a specified period of time. They are also instructed on the practice of writing pseudo-code when developing algorithms. The students are then asked to design the algorithm, describe it with pseudo-code, implement the program and then test it on the board.

In a follow-on activity, students are taught the FOR loop programming construct. They are then introduced to the basic operating princples of servo motors and are then given a datasheet for a very simple motor. Students are also provided a set of functions that allow them to position the servo motor at one specific angle. Students are told that the servo represents the fan motor in our home automation scenario. Students are then tasked with making the servo motor spin when the temperature reaches a specified threshold. This requires that the students apply what they just learned about looping and write a loop themselves that iterates through multiple angle settings of the servo. Figure 9 shows a student's completed circuit design, including the microcontroller, LED, temperature sensor and servo.

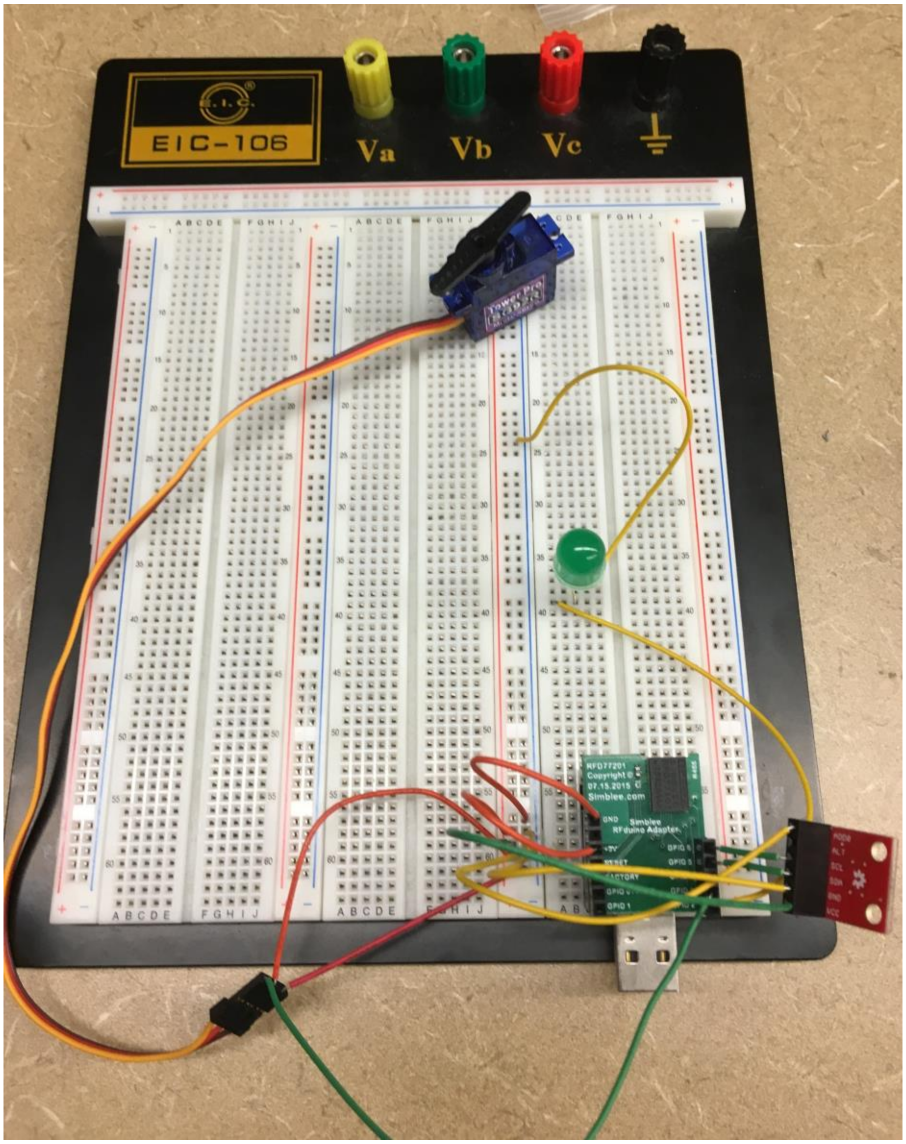

Figure 9. A student's completed hardware circuit design incorporating on to a prototyping breadboard the microcontroller, LED, temperature sensor and servo motor

3.2.4 Wireless networking with Mobile Devices and Graphical User Interfaces

The next major design challenge is for students to incorporate the ability for the microcontroller to wirelessly transmit/receive data to/from a mobile device and then eventually to the "cloud". This requires that the students make use of the Bluetooth and 802.11 networking capabilities of both the microcontroller and the mobile device (smartphone or tablet computer). This also requires that students create a graphical user interface so that the exchange of data can be controlled by a user.

Most students come into the workshop having had experience using a graphical interface (GUI), thus the final part of the workshop begins with GUI development. One aspect of the Simblee microcontroller platform that is particularly appealing for use in this context, a high-school level workshop, are its capabilities to create hardware controlling GUIs from within the embedded microcontroller development environment. Students must first download a provided host application on to their mobile device. The GUI source code that students write is stored on the microcontroller and then retrieved from the microcontroller via Bluetooth when executing. Therefore, students need not be concerned with the typical challenges that come with GUI development for mobile platforms. This enables students to create IoT mobile device applications just by using very simple function calls that are made alongside their microcontroller code for interfacing, sensing, etc., using the same skills they developed during the course of the workshop.

The wireless networking and graphical user interface in this context are complementary, students cannot test and verify the operation of one without the other, therefore these components are presented together. Students are given a brief explanation of Bluetooth networking. The details of the Bluetooth protocol are not discussed, but its distinguishing features are explained in a simple manner. Specifically, the use of Bluetooth in short range, low power applications and a high-level discussion of its master-slave communication structure. Students are provided with functions for advertising their microcontroller as well as for transmitting and receiving data. At this point, before the GUI is fully developed, students are asked to simply advertise their device. An example of the Bluetooth advertisement from multiple student groups being picked up by a smartphone is show in figure 10. Students are asked to observe changes in the received signal strength indicator reading (RSSI) as they move about the room and away from the device

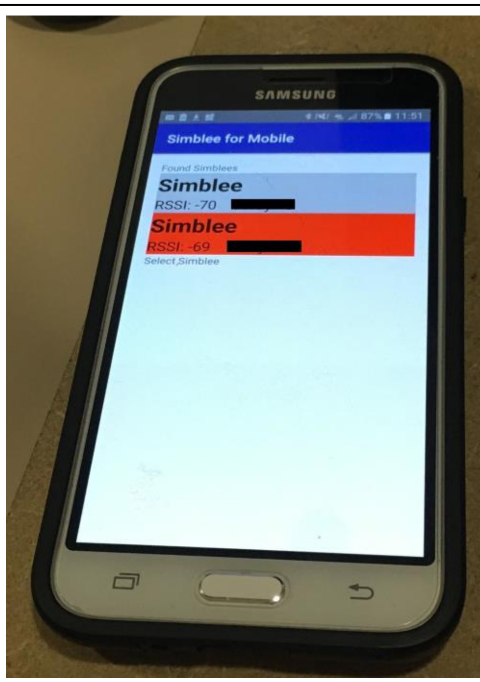

Figure 10. Smartphone picking up advertisement of student projects over Bluetooth network

For the next activity, students are asked to modify their program so that they can read temperature data from the device and display it to the screen. Students are provided function calls for transmitting Bluetooth data as well as for drawing shapes and displaying text to the mobile device screen. At this point in the workshop, students have gained enough experience that providing them with new functions to use is not problematic as long as they are accompanied with a small example. When creating the graphical user interfaces, new information explained to students includes the mobile display's graphical coordinate system, how RGB values map to colors and GUI callback functions. After being provided with that information and examples, are equipped to create their own unique graphical user interface. Figure 11 shows is an example GUI designed by one of the student participants. The user interface displays the temperature from the sensor and the student included a red bar that grows in proportion to the ambient temperature. A button that remotely controls the LED was also included. Figure 12 shows a student using their custom-developed mobile application to control and monitor their project.

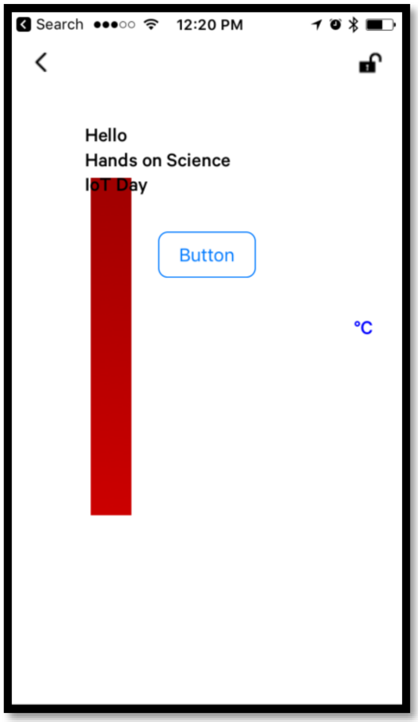

Figure 11. A sample graphical user interface designed by a participating student. The GUI can remotely control an LED, as well as display information from the sensor

For the final activity, students are taught the meaning of the internet "Cloud" and its relevance to IoT is explained and demonstrated to the students. Since this activity is designed to be only a few hours long, and so as to ensure that all students get exposed to all of the primary facets of IoT, regardless of whether or not they finish the tasks, the cloud component is primarily demonstrated to the students. After the demonstration an explanation is given to students on how their mobile device is able to act as a 802.11 internet gateway for their Bluetooth connected IoT device. Students are then shown how each wireless device has a unique electronic serial number (ESN) that can be used to identify it on the Cloud platform. Using a pre-assembled, functional design, a walkthrough of registering the mobile device on to the Cloud service is given. Students get to see how the same ESN viewed earlier is used to address a specific device. A simple JavaScript webpage indicating a pre-set temperature value is used as the information source for the device. In the final demonstration, the fan motor is made to come on based on a comparison of temperature set on the cloud-hosted web page and the sensor detected by the device. Sample code is given to the students so that those who wish to try to interface their project to the cloud can do so, thereby completing the loop of going from a sensor, through multiple wireless networks to the cloud and then back to the device. Table 2 shows a summary of all the workshop's activities.

Figure 12. Student testing the final circuit and mobile application

Table 2. List of all activities

#	Description
1	Assemble Microcontroller / Breadboard
2	Examine Sensor Datasheet
3	Create Sensor-Microcontroller Schematic
4	Configure Integrated Development Envionment
5	Write code to print pin numbers to debug terminal
6	Modify function call for reading sensor data
7	Develop algorithm (Pseudo-Code) for LED control
8	Implement and test temperature controlled LED
9	Integration of Servo motor
10	Bluetooth Advertising and data transmission
11	Display temperature data to mobile device screen
12	Incorporate buttons to Graphical User Interface
13	Internet cloud connectivity demonstration

4. Results

In this section results from the workshop offering are presented. Results are presented in four forms: participant surveys, instructor interviews, classroom observations and direct assessments of student work. To date, the workshop has been offered twice. The majority of results presented here, unless otherwise noted for comparison purposes, were taken from the most recent, improved offering.

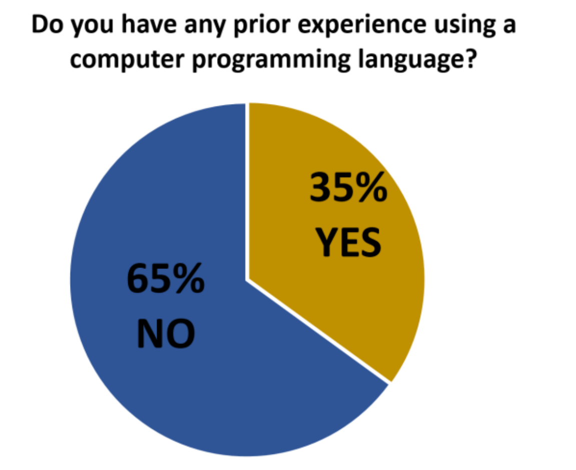

Figure 13. Pre-activity survey of student experience

The workshop was geared towards high school students. There were 20 student participants and all 20 students were in the 11th grade at the time of the workshop. The students were a part of the host University's school of Engineering college prepatory program. In advance of the event, the students were polled and asked about their prior experience. Since a large portion of the workshop requires the students to write code, the students were asked: "Do you have any prior experience using a computer programming language?" Only 35% of participants reported having any previous experience developing software, indicating that the majority of students would be writing their first ever computer program during the activity (figure 13).

The students were also asked about their preferred field of engineering (figure 14). The workshop was designed to encourage participation in engineering in general, but specifically highlight electrical and computer engineering. The survey shows that 25% of the students

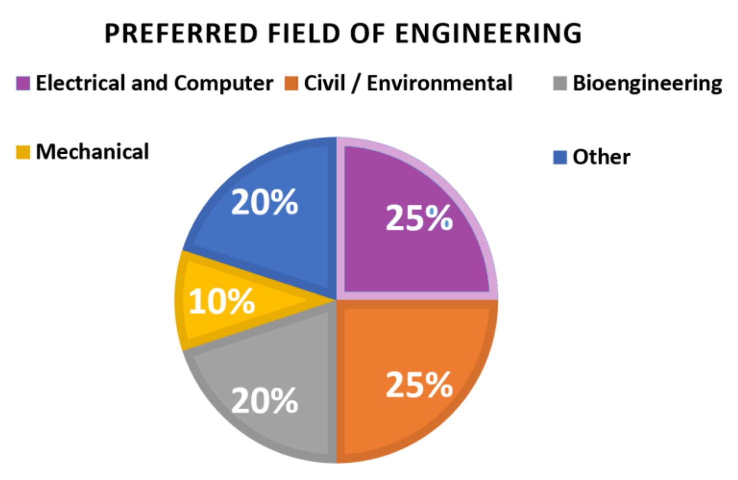

Figure 14. Pre-activity survey of student engineering interests

coming into the workshop already had an interest in electrical and computer engineering, while the remaining 75% were interested in other engineering fields. This mix of students provides a good measure of the degree to which we can reach students with this activity who may not be particularly interested in just electrical and /or computer engineering.

4.1 Student Survey Results

Figure 15 and table 3 show the results from the first survey request, where students were asked to rate the degree to which they learned new information. The results show that 70% of participating student rated the amount of new information they learned as high or very high.

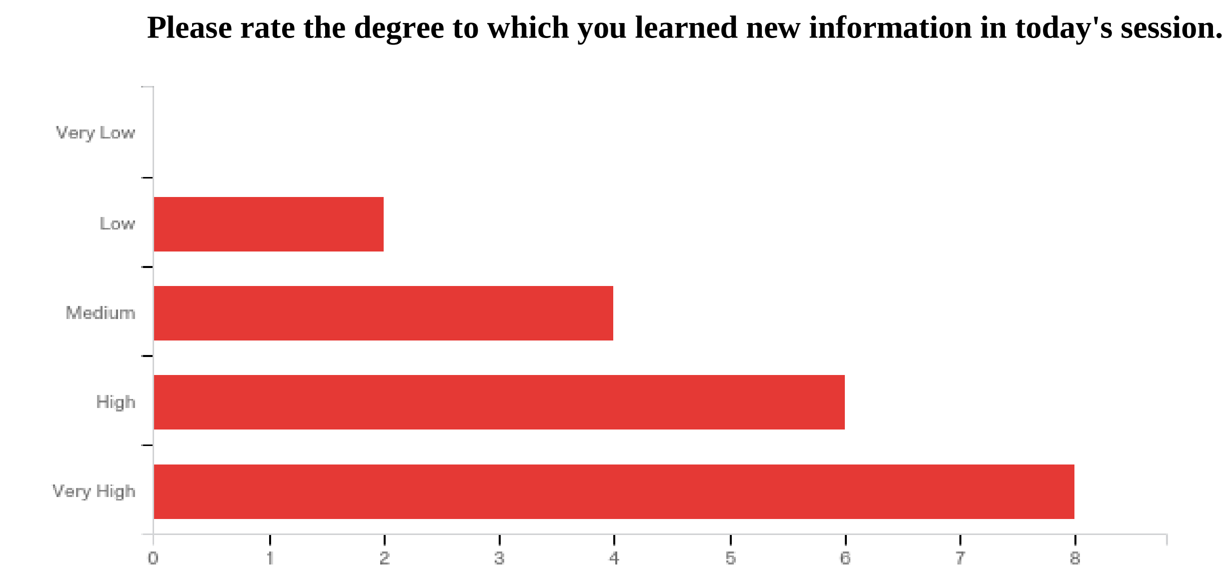

Figure 15. Survey Question #1

Table 3. Survey Question #1 Summary

Answer	%	Count
Very Low	0.00%	0
Low	10.00%	2
Medium	20.00%	4
High	30.00%	6
Very High	40.00%	8

Figure 16 and table 4 show the results from the second survey request, were students were asked to rate the degree to which they learned new information. The results show that 75% of participating students found the activity very engaging, rating this aspect as high and very high

Please rate the degree to which the information was presented in an understandable, interesting and engaging manner.

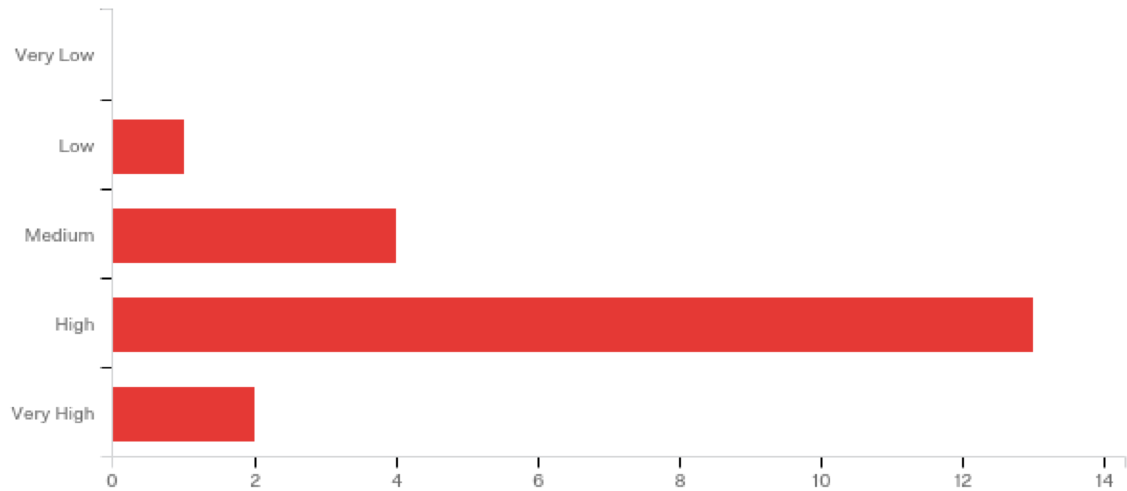

Figure 16. Survey Question #2

Table 4. Survey Question #2 Summary

Answer	%	Count
Very Low	0.00%	0
Low	5.00%	1
Medium	20.00%	4
High	65.00%	13
Very High	10.00%	2

Figure 17 and table 5 show the results from the third survey request, were students were asked whether or not they would like to participate in another event on IoT. The results show that 75% of participating students enjoyed the activity enough that they indicate that they would want to participate again at a future time.

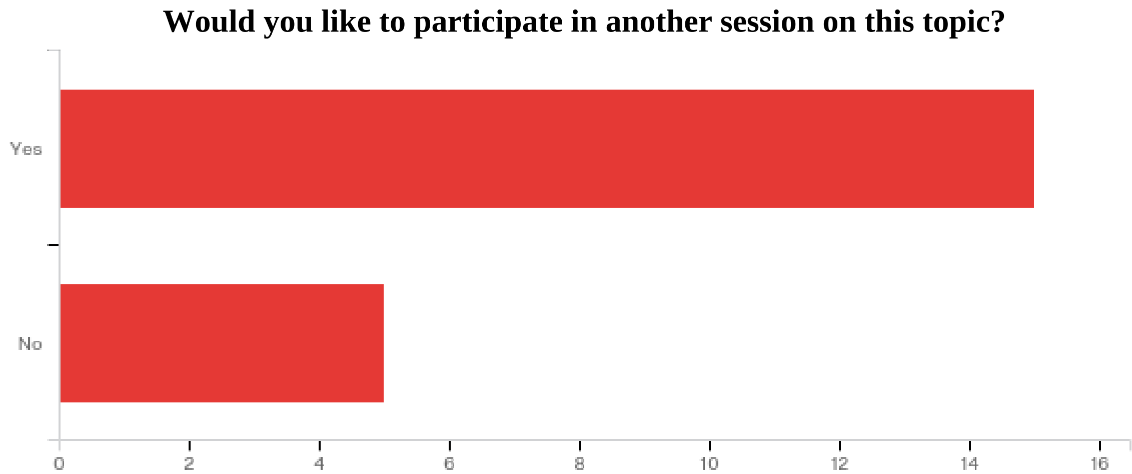

Figure 17. Survey Question #3

Table 5. Survey Question #3 Summary

Answer	%	Count
Yes	75.00%	15
No	25.00%	5

Figure 18 and table 6 show the results from the fourth survey request, were students were directly asked to rate the degree to which the activity increased their interest in specifically Electrical and Computer Engineering. The results show that 60% of participating students looked at electrical engineering more favorably as a result of the event. This is a very positive result as only 25% of students specified electrical and computer engineering as their field of choice going into the activity.

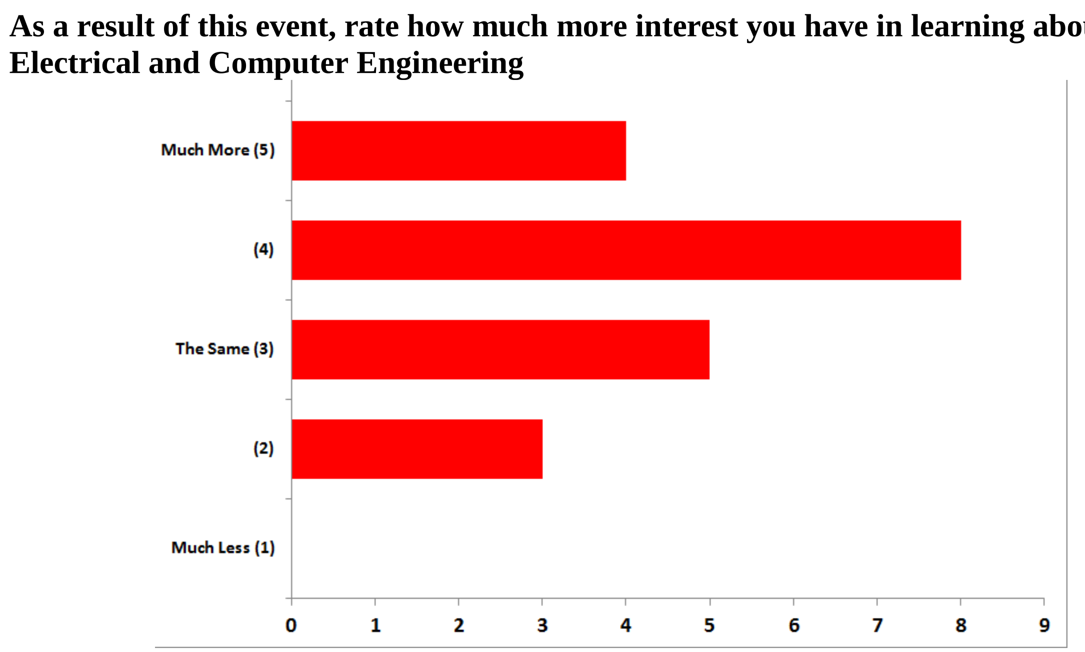

Figure 18. Survey Question #4

Table 6. Survey Question #3 Summary

Rating	Count
Much Less (1)	0
(2)	3
The Same (3)	5
(4)	8
Much More (5)	4

Figure 19 and table 7 show the results from the fifth survey request, were students were directly asked to rate the degree to which the activity increased their interest in learning about the Internet-of-Things. The results show that 65% of participating students had their interest piqued.

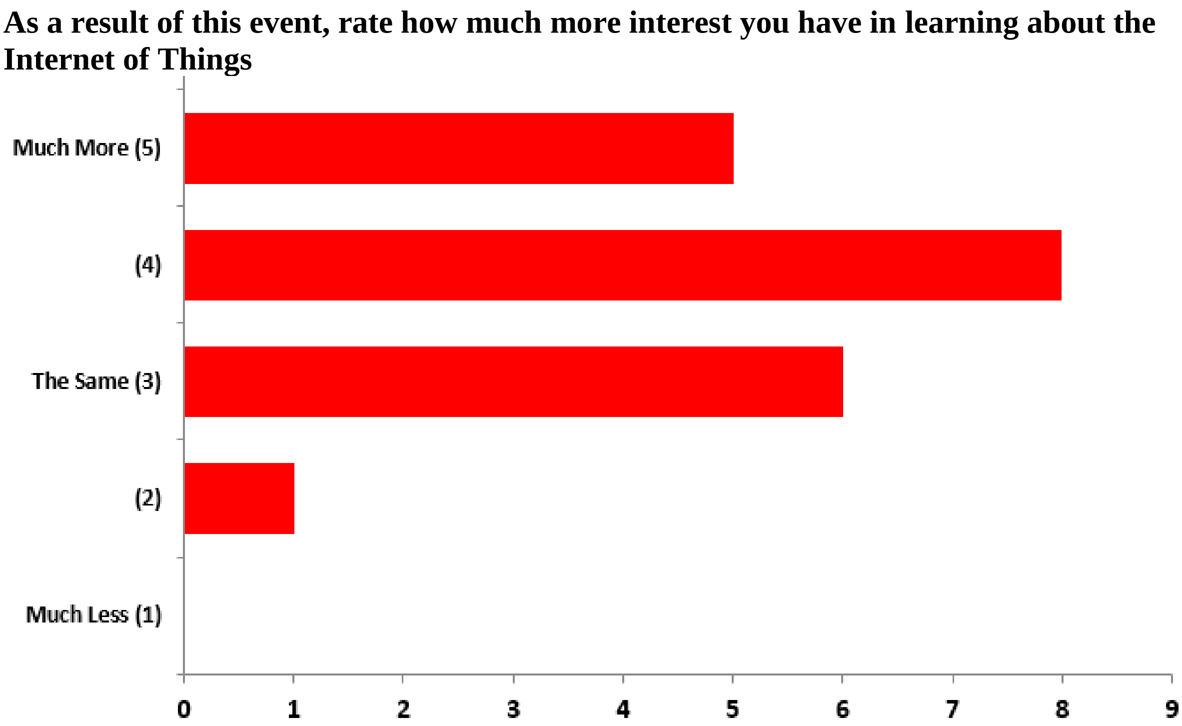

Figure 19. Survey Question #5

Table 7. Survey Question #5 Summary

Rating	Count
Much Less (1)	0
(2)	1
The Same (3)	6
(4)	8
Much More (5)	5

4.2 Classroom observations

A classroom observation protocol was used to obtain a measure of student engagement. The observation protocol employed was loosely modeled after the COPUS, the classroom observation protocol for undergraduate STEM (Smith et al., 2013). An observer was asked to make evaluations of the level of engagement about the groups immediately surrounding them, rating them as high, medium or low. A rating of high indicates that the students were fully involved at the task at hand, where as a low rating indicates that the students were not involved or not interested in the current task. The observer sampled the level of engagement at uniform intervals, once every 30 minutes. It is important to note that these observations are not reflective of the state of the entire class, but rather the observer's opinion and general feel of the few groups immediately surrounding them. The plot of figure 20 shows the summarized result of the observer's ratings from the initial offering of the activity. As with many activities of this type, interest in the activity started off high and then dropped off significantly as time went on, and near the end many groups had lost interest.

One of the observed groups that appeared to quit on the activity was asked directly on how the activity could be modified in order to keep their interest for the entire time. The response from one student was: "It would have been a lot more interesting if we could control everything from my phone". The initial workshop did not contain a component that involved using their mobile phone. This comment led directly to the inclusion of the graphical user-interface component, which was enjoyed by many in the second offering of the workshop.

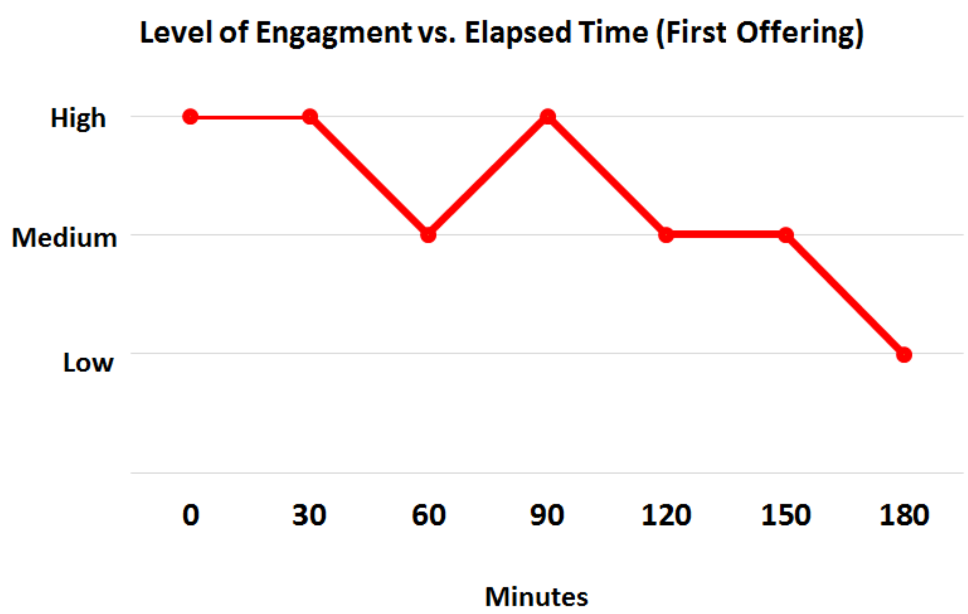

Figure 20. Summary of classroom observations from first offering of workshop

4.3 Direct assessment of student work

As a form of direct assessment, samples of student work were analyzed. Specifically, the source code from each group was collected and later analyzed. Student involvement and completion of the workshop was assessed by using the quality of the source code as a performance indicator. The collected source code samples were evaluated and completeness was determined by mapping the code back to the activities listed in Table 2. Table 8 shows the rubric used in the assessment, ranging from unsatisfactory (no tasks completed and/or code does not compile) to exemplary (all major tasks completed, up to task #11, development of the graphical user interface on the mobile device).

Table 8. Direct assessment rubric

	Unsatisfactory	Developing	Satisfactory	Exemplary
Workshop Completion (assessed from source code)	No tasks completed, code and /or code does not compile	Some tasks completed, up to and including reading from sensor (#6)	Most tasks completed up to and including integration of Servo (#9).	All tasks completed, functional, up to and including GUI development (#11)

Table 9. Direct Assessment Results

Assessed Level	Count (2 Students Per Group)
Unsatisfactory	**1**
Developing	**2**
Satisfactory	**5**
Exemplary	**2**

Table 9 shows the assessment results. The results show that 70% of students performed at a satisfactory or above level, 20% of students completed the workshop in its entirety while 30% of students were evaluated to not have performed at a satisfactory level. It is important to note that those who did not finish the workshop assignments in its entirety still benefited, as the workshop was purposely designed to accommodate students with various levels of experience. It is most likely that those who came into the workshop with prior programming experience were the ones that were able to complete the activity.

5. Summary and Conclusions

The administrator of the students (not the instructor) was asked to evaluate written portion of the student feedback surverys and select comments of interest. The administrator selected the following comments from the following survery request: "*List any positive actions you would take in school based on what you learned today"*

"I'm gonna take a coding class next year in school"

"This activity in general opened me up to a whole new world of knowledge."

" I will be more open minded to learn new things in school related to engineering and science in general."

These quotes show that the activity was successful in achieving its goals. When asked how the activity could be improved, some of the responses included:

"Provide a list of vocabulary"

"More templates, but not too much"

"Slow down the pace since a lot of people were behind"

"Maybe slow the pace down"

These suggestions provide helpful feedback for improving future offerings. While not all exactly the same, they all point towards a similar issue. The workshop has so much content packed into one event, and so much of it is new to the students, that students can easily get lost and therefore need either more time or more aids. In the future, more time will be dedicated to the development of the graphical user interface, as students seem to enjoy that aspect the most. Leaving more time for that portion will in turn require shortening other tasks. The instructor feels that the point at which student groups diverged was integration of the servo motor, which required students to develop loop, some for the first time. For future offerings, it is recommend to leave out this portion, as the main aspects of IoT can be highlighted to students without it, thereby leaving more time for other activities, and improving the percentage of students who complete all of the activities.

In summary, an engaging workshop designed to introduce students to the Internet-of-Things was presented. The goals of the activity were threefold, build excitement for electrical and computer engineer, IoT and provide a foundation for developing IoT technologies. The results show that the workshop was successful in achieving those goals.

References

Feaster, Yvon , Jiannan Zhai, and Jason O. Hallstrom. 2015. Serious Toys: Introducing Sensors and Sensor Networks in Pre-collegiate Classrooms. In Proceedings of the 16th Annual Conference on Information Technology Education (SIGITE '15). ACM, New York, NY, USA,

Gartner, Inc. (November 2015). Gartner Says 6.4 Billion Connected "Things" Will Be in Use in 2016, Up 30 Percent From 2015. Retrieved from <http://www.gartner.com/newsroom/id/3165317 > on February 10, 2017.

NXP. I2C-bus specification and user manual. Retrieved from <http://www.nxp.com/documents/user_manual/UM10204.pdf> on February 10, 2017.

Smith, M., Jones, F., Gilbert, S., & Wieman, C. (2013). The Classroom Observation Protocol for Undergraduate STEM (COPUS): A New Instrument to Characterize University STEM Classroom Practices. *CBE-Life Sciences Education*, *12*(4), 618-627.

Simblee. Retrieved from <https://www.simblee.com/> on February 10, 2017.

Texas Instrucments. TMP102 Datsheet. Retrieved from <http://www.ti.com/product/TMP102> on February 10, 2017.