Teaching the Internet of Things (IoT) Using Universally Available Raspberry Pi and Arduino Platforms

Prof. Gary J. Mullett, Springfield Technical Community College

Gary J. Mullett, a Professor of Electronics Technology and Co-Department Chair, presently teaches in the Electronics Group at Springfield Technical Community College (STCC) located in Springfield, MA. A long time faculty member and consultant to local business and industry, Mullett has provided leadership and initiated numerous curriculum reforms as either the Chair or Co-Department Chair of the four technology degree programs that constitute the Electronics Group. Since the mid-1990s, he has been active in the NSF's ATE and CCLI programs as a knowledge leader in the wireless telecommunications field. A co-founder of the long running National Center for Telecommunications Technologies (then the ICT Center) located at STCC, Mullett also played a principle role in the development of the innovative and long running Verizon NextStep employee training program. The author of two text books, Basic Telecommunications – The Physical Layer and Wireless Telecommunications Systems and Networks, Mullett did both his undergraduate and graduate work (in Remote Sensing) in the ECE Department at the University of Massachusetts at Amherst where he also taught the undergraduate sequence of courses in electromagnetics. He has presented at numerous local, regional, and national conferences and also internationally on telecommunications and wireless topics and on the status of the education of electronics technicians at the two-year college level. His current interests are: the development of novel and innovative systems-level approaches to the education of technicians, applications of the emerging field of wired and wireless networked embedded controllers and sensor/actuator networks, and cyber-physical system applications in the context of the Internet of Things (IoT).

Teaching the Internet of Things (IoT) using universally available Raspberry Pi and Arduino platforms

Abstract

Major technology corporations like IBM, Cisco, Microsoft, Amazon, Google and others have taken direct aim at the rapidly developing Internet of Things (IoT) paradigm. These corporations believe that the next major technologic evolution (and hence their future markets and customers) will revolve around this newest application of the Internet. These companies and most involved in the technology fields believe that we are on the cusp of the next transformation of machinery and infrastructure: the process of adding intelligence and connectivity to small to large-scale technology and infrastructure systems and in this process creating the Internet of Things. Already, academic and industry experts in various technical fields have given catchy names to these proposed systems: autonomous cars, Smart Grid, Smart Buildings, Smart Homes, e-health care, are but a few terms that have made it into the popular press. These large-scale and not-so-large-scale applications are becoming possible due to the convergence of several key enabling technologies. Essentially, through the use of networked embedded controllers (known as ambient intelligence) and complex wired and wireless sensor and actuator networks one is able to create intelligent infrastructure systems (i.e. cyber-physical systems) that have the potential to change almost every aspect of mankind's interaction with the environment and most all other forms of human commerce and endeavors.

Presently, formal education in these innovative Internet applications is lacking. Cisco, through its online Networking Academy offers a short overview course about the Internet of Everything (IoE) and has announced its intent to offer more online courses about the topic. However, access is restricted to colleges that belong to the Cisco networking academy program. IBM has recently launched its Internet of Things Foundation that offers business and industry partners, as well as, educational entities, development tools to implement and test out their IoT applications with the further ability to visually display acquired data in real time. However, as of yet, most colleges that offer engineering technology education at the two-year college level, do not offer courses or programs to introduce and teach this new technology. This paper will present our experience in our initial attempts to impart an introduction of this technology to our students and to also possibly induce students to become interested in STEM fields. A new four credit course ELE-111 (three hours of lecture and three hours of lab), entitled, Internet of Everything, has been offered to our students this past year. This paper will recount our experience with this specifically hardware centric hands-on course. Course and lab content will be covered and our experience with what worked and what did not will be discussed. This freshly renewed effort to graduate technicians with the skill sets needed to install, evaluate, maintain, and up-grade these IoT systems as they are envisioned is destined to be an ongoing process at our institution.

Introduction

By this time, most have heard of the popular culture moniker "Internet of Things" or IoT. Many large technology corporations have begun taking direct aim at this rapidly developing technology area. Microsoft, IBM, Cisco, Goggle, Apple and many other enterprises believe that the next major technologic evolution (and hence their future markets and customers) will revolve around this newest application of the Internet. These companies and most involved in the technology fields believe that we are on the cusp of the next technologic evolution: the transformation of machinery and infrastructure - the process of adding both intelligence and connectivity to small to large-scale technology and infrastructure systems. The end result of this process of crafting innovative, intelligent, system applications is the creation of a space known (collectively) as the Internet of Things. Already, academic and industry experts in various technical fields have given catchy names to these proposed systems: Smart Grid, Smart Homes, Smart Buildings, Smart Cities, Smart Manufacturing, autonomous and connected cars, e-health care, and microgrids are but a few terms that have made it into the popular press. As one can easily surmise, the key term here is “Smart” but just as important is the fact that these systems also enjoy connectivity via modern communications technology (i.e. computer/data networking). These large-scale and not-so-large-scale applications are becoming possible due to the evolution and convergence of several key technologies. Essentially, through the use of networked embedded microcontrollers (ambient intelligence), connected complex sensors and actuators (sensor networks), and wired and wireless networking one is able to create intelligent infrastructure systems (i.e. cyber-physical systems) that have the potential to change almost every aspect of mankind’s interaction with the environment and most all other forms of human commerce and endeavor. The details of this technology evolution have been chronicled elsewhere by this author and others[1-9] and will not be repeated here.

Presently, formal technology education in these innovative Internet applications is wanting. Certainly, there is no shortage of attention to the underlying technologies at the upper-level undergraduate and graduate schools of four-year engineering colleges. In fact, many research institutions are devoting their efforts to these new technology fields and possible applications of the IoT. However, most colleges that offer engineering technology education at the two-year college level, as of yet, do not have educational courses or programs to introduce and teach this new technology. Recently, Cisco, through its online Networking Academy, has started to offer a short overview course about the Internet of Everything (IoE) and has announced its intent to offer more online courses about the topic. This is a well thought-out business decision driven by Cisco’s desire to gain market share of the networking piece of the IoT. However, access to this material is presently restricted to colleges that have Cisco networking academies. Also, IBM has recently launched its Internet of Things Foundation[10] that offers business and industry partners, as well as, educational entities, development tools to implement and test out their newly designed IoT applications with the further ability to visually display acquired data in real time.

The Problem

Ironically, some of the very technologies that are driving this newest technology area have in some cases, for decades, been trying to remove/detach the hardware from the user. To provide computing power to the masses, the computer industry has attempted to shield the user from the intricacies of both the computer hardware and software needed to provide computing power. Often this strategy, to avail a technology or family of products to the masses, is known as proving a “turn-key” solution. Certainly this has been an effective approach, as evidenced by the success and proliferation of the personal computer (PC), the cellular telephone or smartphone, and the automobile to name but three highly technical products designed for mass consumption by mankind. At the same time, providing a turn-key product/commodity has in many instances reduced the ability of the populace to “tinker” with and become familiar with the underlying technology. This last fact is not entirely due to the manufacturer’s desire to provide a turn-key solution but in the case of the three examples just mentioned is also a natural consequence of the evolution of electronics technology[11,12,13] and the seemingly never-ending micro-miniaturization of electronic systems (i.e. Moore's Law). One of the consequences of this is trend is the uneconomical cost of repair for the electronic sub-systems in these products and what we tend to define as "throw-away" or disposable technology when considering consumer electronics products. Computer software has likewise followed a similar trajectory in its evolution. From the earliest programming languages with their oftentimes complex syntax and coding structures to today's point-and-click graphical programming software (GPS), we have developed computer operating systems like the venerable "Windows" series with graphical user interfaces (GUIs) to allow even the most neo-type computer user to rapidly become a capable operator of a PC. Today's automobiles run millions of lines of computer code during their operation and our only chore to get them operational is to turn the key on (or press the start button), step on the gas petal, and turn the steering wheel. Seemingly, our only interaction with the automobile's software is when the infamous "Check Engine" annunciator light comes on. According to most of the major automobile manufacturers, in the not too distant future, self-driving, autonomous cars will further reduce our interaction to just getting in and specifying a destination! Again, it has all been about bringing the technology to the masses.

Technology education tends to be aimed at giving the required skills to the technician that will allow them to deal with technology based systems either during the manufacturing process or final test or in the field after the product has been placed in use (the after-market). The job market for technicians tends to be the driving force behind curricula change in the two-year college technology education arena. Over the course of the last three decades in particular, the transformation of the electronics industry and the computer industry (from mainframes to PCs) has changed the face of technician training at the two-year college level. In the electronics area, programs have been discontinued due to low enrollment or sometimes morphed into computer technology programs with an emphasis on networking. In many cases, these morphed programs have given up computer hardware topics in favor of software (i.e. operating systems and programming languages) and in numerous instances they have adopted the Cisco Network Academy program in their quest to remain viable. In any case, these programs (both electronics technology and those that are/were called computer technology) typically do not cover all of the enabling technologies of applications of the Internet of Things.

At Springfield Technical Community College (STCC), Electronics Technology was a legacy program that was implemented at the inception of the college almost fifty years ago. Today that program is known as Electronic Systems Engineering Technology (ESET), is ABET accredited, and struggles with low enrollment. The Springfield, Massachusetts area where the college is located has lost most of its electronics manufacturing base and graduates of the program tend to be hired as the sole electronics tech for small local companies or as an electronic tech by some of the laser/photonics manufacturers (a developing industry base) in the surrounding area. The curriculum for the ESET program has been transformed to include sensor networks, wireless technologies, and some computer networking along with a concentration on instrumentation and embedded controllers (covering most of the enabling technologies of the IoT). The Computer Systems Engineering Technology (CSET) program at STCC used to be known as the Computer Maintenance Program when first offered in the early 1980s and Digital Equipment Corporation (DEC) was headquartered in Maynard, Massachusetts and had a large manufacturing presence in western Massachusetts. Times have changed and DEC is no more. Today, CSET is a very popular program that is associated with the Cisco Networking Academy program. Graduates receive educational training on computer networking, operating systems, programming, PC repairs (i.e. A+), computer security, and server systems. Most graduates get employed locally by enterprises that need PC and networking support services. If one was to refer to the seven-layer ISO networking model, the CSET program could be best described by saying that it concentrated on the bottom four layers of that seven layer model.

However, as pointed out previously, the computer industry, either intentionally and/or through the help of ongoing technology evolution, has attempted to shield computer users from the intricacies of computer hardware and software. At the two-year college level, this is nowhere more evident than in the Cisco Network Academy's coverage of computer networking. Since its inception in 1996, Cisco's networking program for educational institutions has been aimed at Enterprise (business) applications for networked computers. The IoT paradigm changes the networking dynamic by introducing applications that involve embedded controllers and non-business type functions (i.e. control operations, infrastructure operation and monitoring, machine to machine communications, etc.). Presently, the Cisco Networking Academy does not require any understanding of computer hardware internal workings (let along embedded micro-controllers) for one's successful completion of its course material and the attainment of a Cisco Certified Networking Associate (CCNA) certification. Furthermore, there are no technical prerequisites required by Cisco to take their introductory networking course. Of course, this was the original vision of George Ward the architect[14] of the Cisco Networking Academy program. He felt that anyone should be able to become a network technician. Another example of this philosophy is the on-line, PC Pro course by TestOut.com[15]. This course, designed to prepare the student for A+ certification (PC repair), omits any information about the internal workings of a computer or explanation of how information is represented or flows in a microprocessor or microcontroller. This author has long maintained that technicians that deal with IoT applications and cyber-physical systems will need a much different skill set than those of a networking or PC technician.[16,17,18]

Tangentially, but relevant to this present situation, there has been a growing "makers" movement which involves people who have a desire to make projects or create new things. In a different time, electronic hobbyist could build crystal radios or put together more sophisticated electronics kits including complete professional grade superheterodyne shortwave radios by Knight Kit and others that rivaled the Hallicrafter brand in quality. Or, one could tune-up your own automobile or build a remote control (RC) airplane. As technology became more complex and ICs replaced discrete components much of this type of electronics tinkering activity faded away except for the very dedicated hobbyist. Recently, through web sites, blogs, and other social media, people interested in this type of activity have been sharing experiences and information about a host of possible projects and technologies that until recently where beyond what one individual had access to. Whether or not one considers this movement a fad or whatever, it has gained a life of its own and seems to be growing. It appears that there are those among the general population that still like to build things, tinker, and yes - hack. Now with extremely powerful, low cost electronic processing readily available, low cost 3D printers, low cost motors, inexpensive color cameras, solid-state Lasers and high power LEDs, drones, new wireless technologies, etc. and instructions readily available on the Internet, the hobbyist has been empowered to become the entrepreneur. Of course, this is in juxtaposition to the long time stance of our present computer education at the technician level. If one wants to learn about microcomputer architecture you must attend a four-year engineering school or find a two-year technical college that offers a course on embedded controllers or go to a micro-controller manufacturer's web site and learn it on your own.

Lastly, one would have to be totally off the grid to not be aware of the tremendous push for science, technology, engineering, and mathematics (STEM) education. Since many of the jobs to be created in the future are dependent upon knowledge in these areas, there has been a great deal of support from the National Science Foundation (NSF) for initiatives and special programs to boost the interest of students in these topics and to influence them to pursue careers in these areas. This is not just a United States undertaking, it is a world-wide effort being undertaken by all the industrialized nations. Recently, new NSF programs like ITEST, IUSE, and STEM-C[19] have called for proposals that attempt to get young people and under-represented minorities and females interested in STEM topics and computer coding. On January 30th of this year (2016), President Obama announced a $4.2 Billon educational initiative called "Computer Science For All". This proposal is a plan to reboot computer science education programs in what the White House is calling an "Eisenhower moment" for technical education. This three-year initiative would help train teachers, equip classrooms, and develop new class materials[20]. All this in an effort to increase student knowledge about the computer and how to apply it to solve problems and perform useful functions.

A Solution?

Recently, the CSET program at STCC has introduced a new course to fill a perceived need for additional technical skills related to the IoT and to fill an apparent yearning for knowledge about applications of microcontrollers enabled by the Internet. This new three credit course, ELE-111 – The Internet of Everything and its one credit associated lab, ELE-111L have been offered each semester during the past 2015-2016 academic year (see www.stcc.edu for the course and lab descriptions). Each semester the course and lab have been over-subscribed and students have had to be turned away. This course consists of a presentation of: microprocessor and microcontroller hardware theory, a software programming language, operating system concepts, and details of interfacing the controllers to the real world. The lab consists of hands-on exercises that provide the student with the opportunity to program the hardware to perform some simple useful control function(s) and to make projects that can be controlled through the Internet. To implement the course the faculty choose to use low-cost universally available hardware that is available from a variety of online sources. The microprocessor platform utilized is the Raspberry Pi and the associated open-source microcontroller platform is the Arduino Uno. The Arduino Uno shown in Figure 1 below has been around for some time and has been cloned by numerous vendors. Due to its longevity, there are many accessories (known as shields) available from the same vendors. A quick search of Amazon.com will provide one with many different Arduino boards, Arduino starter kits, and Arduino shield options. Furthermore, there are many online resources for apps or software programs pre-written for the Arduino. One may download software code to the Arduino board through its USB port from another device.

This board, with a suggested cost of approximately $25, uses an 8-bit microcontroller, has 14 digital input/output pins (of which, 6 can be used as Pulse Width Modulation (PWM) outputs, 6 analog inputs, a USB connection, and a power jack. With the various digital inputs and outputs and the analog inputs the Uno can be used to control external hardware much like the embedded controller of a typical electronic system. Furthermore, the Uno is fairly easily interfaced to the Raspberry Pi board. There are a host of accessories available to be used with the Uno and there are more powerful Arduino boards (like the Zero) that utilize a 32 microcontroller, however it is roughly twice as expensive as the Uno. A search of Amazon.com yields a selection of over one thousand Arduino products with various “starter kits” and accessories.

Figure 1 – Microcontroller Platform – Arduino Uno

The other hardware platform used in the course is the Raspberry Pi 2 Model B starter kit. This device has a basic cost of approximately $50 and essentially is a full blown PC on a board. One needs to provide a monitor and keyboard and you are good to go. The Raspberry Pi 2 platform shown below in Figure 2 is the second generation Raspberry Pi.

Figure 2 – Microprocessor Platform – Raspberry Pi 2 Model B

It has a 900 MHz quad-core ARMv7 processor and 1 GB of RAM memory. It comes with 4 USB ports, 40 GPIO (general purpose Input/Output pins, full HDMI and full Ethernet ports, combined audio jack and composite video jack, camera and display interfaces, micro SD card slot, and 3D graphics core. Because it utilizes an ARMv7 processor it can support: Linux, Snappy Ubuntu, and Windows 10. It can also run the Windows 10 IoT Core from Microsoft. This is Microsoft's special version of the Windows 10 operating system crafted for IoT applications[21].

The associated Internet of Everything lab course uses the CSET operating systems lab and the embedded controllers lab for hands-on activities and project construction. The operating systems lab has computer monitors with HDMI inputs and the other required hardware to make the Raspberry Pi operational and the embedded controllers lab has basic test and measurement instrumentation available and common electronics hand tools to allow for basic electronics experimentation and project construction. So far, the student response to the course and the lab has been overwhelmingly positive. The following Table lists the course activities for the entire semester. Of course, as with most technical courses, the first time through is the test run and the occasion when one discovers what works well and what does not. The course content will certainly be tweaked after the first run through.

Week	***Topic***
1	Orientation. IOE overview and accounts setup. What is IOE? Basic Electronics; voltage, current, resistance, and power
2	The Cisco IOE course. Intro to IOE. Electronic units, measurements. Using a breadboard. Reading schematics. Using a Digital Volt Meter (DVM)
3	Raspberry Pi basics. Building and testing the Raspberry Pi operating system. Cisco IOE discussion.
4	Raspberry Pi Interfacing. GPIO, GPIO differences (PI 2, PI B+), I2C Bus, SPI. Controlling the input and output pins.
5	Security of IOT. Arduino basics. Comparing the Arduino to the Raspberry Pi.
6	Home automation – a framework to look at IOE and IOE integration. Survey of common products, security concerns, interoperability and user interface problems. Programming the Raspberry Pi
7	Analog and digital interfaces on sensors. I2C and SPI bus commands & sensors.
8	Adding digital sensors to the Raspberry Pi and Arduino. Reading switches. Driving LED's, SPI interfaces to displays, HDMI and touch screens.
9	Networking primer. Raspberry Pi and Arduino networking. IOT Security
10	Networking your IOT devices Logging data from IOT devices to Google Spreadsheets Logging data from IOT devices to networked databases Cloud based IOT data collection and control Using a smartphone or tablet for a display or the control of IOT devices
11	Programming the Raspberry Pi. Programming the Arduino. Finding existing code or libraries for the Raspberry Pi. Using GIT version control system.
12	Control system primer. Measurement accuracy and sampling frequency.
13	IOE system integration Projects. Examples include temperature, humidity and environmental monitoring, GPS/Position, etc.
14	IOE system integration. Logging to or control from remote servers, cloud or smart phones
15	Review and Finals

Table 1 – Internet of Everything Course Topics

As one can see from the table, the course starts with a quick overview of the general concepts of the IoT. Since there are no prerequisites for the course, some time is spent on an overview of basic electronics concepts and components including basic control elements and sensors. Focus then shifts to the Raspberry Pi platform and its operation and capabilities. How one interfaces to the Raspberry Pi is covered in detail including the interfacing of the Arduino platform. Using home automation as a framework to look at IoT integration, sensors and communications are added to the hardware platforms. Networking the platforms and programming/coding the Raspberry Pi and Arduino are covered next. The course finishes with system integration topics and data logging. Since there is no required text for the course, reading assignments, references and reference material, and Internet research on selected topics are provided during the semester.

The lab work for the semester mirrors the classroom topics and was fairly structured during the first two thirds of the course. After that, the students were in full project mode and the lab time was used to build and test their particular project designs. Table 2 below shows the lab schedule for the semester. Some representative lab projects are shown in Figure 3 below.

Week	***Topic***
1	Resistors, power supplies, and DVM voltage measurements
2	Electronic units of measure (V, I, R, P) for DC and AC electronics. Using a breadboard to prototype basic circuits. Integrated circuit pin-outs and connector pin-outs
2	Series and parallel concepts. Build a simple timer to learn package pin outs and schematics.
3	Soldering and de-soldering
4	Building an operating system for the Raspberry Pi Raspberry Pi boot up, testing, and network configuration.
5	Securing the Linux system on your Raspberry Pi. Connecting switches and LED’s to the GPIO pins
6	Arduino configuration and testing. Programming the Arduino. Example: Driving serial addressable RGB “Neopixel” LEDs
7	Testing the I2C and SPI bus connectivity. Attaching I2C and SPI devices to the Raspberry Pi.
8	Using common digital sensors Analog sensors and analog to digital conversion
9	Using the Raspberry Pi camera module Uploading camera images to your account on the cset2 Linux server Motion detection.
10	Network testing and troubleshooting. Wireless network configurations
11	Logging data to a Google spreadsheet Pushing sensor data from IOT devices to a database on a Linux Server Logging data to a cloud server
12	Network security lab. Using Wireshark to monitor 802.11 Using Wireshark to monitor Zigbee and Zwave control signals.
13	System integration of the Raspberry Pi and Arduino monitoring of physical systems using temperature, humidity, light levels detectors, position/GPS, and other sensors).
14	System integration of one or more IOT devices with a Linux Internet server or cloud based system.

Table 2 – Lab Schedule for IoE laboratory

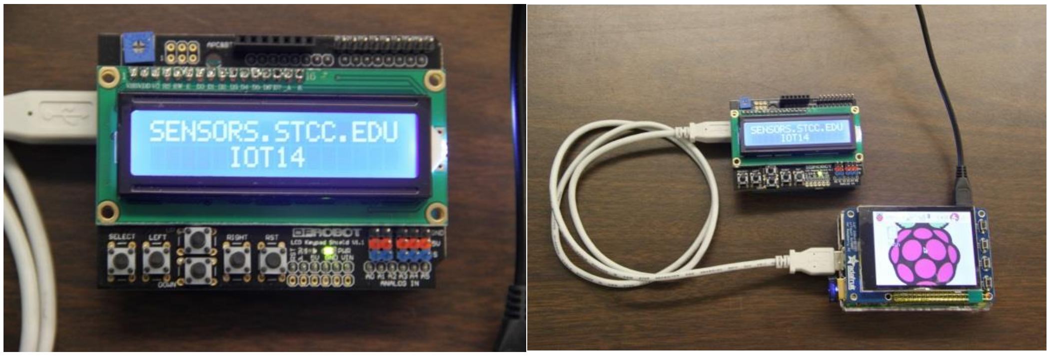

Figure 3 – Representative lab projects with LCD displays (Left – Arduino displaying logging host and ID and Right – Arduino and Raspberry Pi with LCD interfaced together)

Conclusion

After only one semester of experience, it is difficult to draw many meaningful conclusions about the course. However, we are encouraged by the very positive feedback from the students and the fact that they were not in a rush to get the lab done as fast as possible and leave. Many times the students worked on their lab projects outside of the scheduled lab time and were eager to have the lab start! We will have more feedback after this Spring 2016 semester and are entertaining thoughts about running the class during the summer, adding additional sections this coming academic year, and offering the course to non-majors in the hope of generating interest in the STEM fields. The course certainly does not provide all the skills mentioned earlier in this paper needed to deal with the Internet of Things applications nor does it go into the necessary depth to provide expertise about large scale systems. However, it is a start and if it influences students to explore their “maker” side we will believe that in some small measure it has been a success. When this author was growing up there were science fairs, technology was driven by the space race, and there were many opportunities to explore the world with my hands – doing and making things. That opportunity shaped who I am as a person and opened to me many avenues of endeavor in the STEM fields. Even though there is much more advanced technology available today, today’s students do not seem to have as many chances to tinker with and touch it and experience what earlier generations where able to in the technology arenas of the day. Pointing and clicking just doesn’t do it! Hopefully this particular pendulum of popular culture, the “maker” culture, embodied by the Internet of Things concept, is swinging back towards an era when the abstraction of STEM concepts by a young person’s mind starts with making and doing things with readily available, inexpensive technology and of course, the special possibilities enabled by the connectivity of the Internet.

Bibliography

1. *The Disappearing Associates Degree Program in Electronics Technology,* by Louis E. Frenzal Jr. , Proceedings of the 2003 American Society of Engineering Educators Annual Conference and Exposition, Nashville, TN
2. *The New Electronics Technology – Circa 2015,* by Gary J. Mullett, Proceedings of the 2009 American Society of Engineering Educators Annual Conference and Exposition, Austin, TX

3. *Its 2010 and the new Electronics Technology Paradigm is Emerging*, by Gary J. Mullett, Proceedings of the 2010 American Society of Engineering Educators Annual Conference and Exposition, Louisville, KY
4. *Wireless Technologies for e-Healthcare*, IEEE Wireless Communications Magazine, Vol.17 No. 1, February, 2010
5. *Special Issue on Cyber-Physical Systems*, by Radha Poovendran, Krishna Sampigethaya, Sandeep Kumar S. Gupta, Insup Lee, K.Venkatesh Prasad, David Corman, and Jamers L. Paunicka, Proceedings of the IEEE, January 2012 , Vol. 100, No. 1, pp 7-14
6. *Challenges and Research Directions in Medical Cyber-Physical Systems*, by Insup Lee, et al, Proceedings of the IEEE, January 2012 , Vol. 100, No. 1, pp 75-90.
7. *Teaching Networked Embedded Control at the Two-Year College Level*, by Gary J. Mullett, Proceedings of the 2012 American Society of Engineering Educators Annual Conference and Exposition, San Antonio, TX
8. *The Internet of Things (IoT) will create the need for the Cyber-Physical Systems Technician*, by Gary J. Mullett, Proceedings of the 2014 American Society of Engineering Educators Annual Conference and Exposition, Indianapolis, IN
9. *The creation of a Biomedical Engineering Technology program for the 2020s,* by Gary J. Mullett, Proceedings of the 2015 American Society of Engineering Educators Annual Conference and Exposition, Seattle Washington
10. https://internetofthings.ibmcloud.com
11. *The 2010 Gigascale Imperative: Why the instruction of electronics technology must change!*, by Gary J. Mullett, Proceedings of the 2005 American Society of Engineering Educators Annual Conference and Exposition, Portland, OR
12. *Are the Electronics Technology Departments of Today, Destined to Become Academic Service Departments of Tomorrow?*, by Gary J. Mullett, 2006 SAME-TEC Conference, Santa Fe, NM, available on the SAME-TEC web site, Conference Proceedings 2006: www.same-tec.org
13. *Are Today's Electronics Technology Programs Doomed to Extinction or is their Mission Changing?*, by Gary J. Mullett, Proceedings of the 2007 American Society of Engineering Educators Annual Conference and Exposition, Honolulu, HI
14. http://zoominfo.com/p/George-Ward/13874115
15. www.testout.com
16. same as 7.
17. same as 8.
18. same as 9.
19. www.nsf.gov
20. https://www.whitehouse.gov/blog/2016/01/30/computer-science-all
21. https://dev.windows.com/en-US/iot