2023 Annual Conference & Exposition
Baltimore Convention Center, MD | June 25 - 28, 2023
The Harbor of Engineering Education for 130 Years
Paper ID #39791

Enabling Remote Student Learning of IoT Technologies

Dr. Lifford McLauchlan, Texas A&M University, Kingsville

Dr. Lifford McLauchlan is an Associate Professor and Interim Chair in the Electrical Engineering and Computer Science Department at Texas A&M University - Kingsville, and has also worked for Raytheon, Microvision, AT&T Bell Labs, and as an ONR Distinguished Summer Faculty at SPAWAR San Diego, CA. He has over 55 publications covering areas such as adaptive and intelligent controls, robotics, an ocean wave energy converter, green technology, education, wireless sensor networks and image processing. He is a co-inventor on 3 US patents related to control systems. Dr. McLauchlan is a member of ASEE and was the 2012-2014 Chair of the Ocean and Marine Engineering Division. He is also a member of IEEE (senior member), SPIE, Eta Kappa Nu, ACES and Tau Beta Pi, and has served on the IEEE Corpus Christi Section Board in various capacities such as Chair, Vice Chair, Secretary and Membership Development Officer. Dr. McLauchlan has received the Dean's Distinguished Service Award twice and the Dean's Outstanding Teaching Award once for the College of Engineering at Texas A&M University-Kingsville.

Dr. David Hicks, Texas A&M University, Kingsville

David Hicks is an Associate Professor in the Electrical Engineering and Computer Science Department at Texas A&M University-Kingsville. Before joining TAMU-K he served as Associate Professor and Department Head at Aalborg University in Esbjerg, Denmark. He has also held positions in research labs in the U.S. as well as Europe, and spent time as a researcher in the software industry.

Dr. Mehrube Mehrubeoglu, Texas A&M University, Corpus Christi

Dr. Mehrubeoglu received her B.S. degree in Electrical Engineering from The University of Texas at Austin. She earned an M.S. degree in Bioengineering and Ph.D. degree in Electrical Engineering from Texas A&M University. She is currently an associate professor in the School of Engineering and Computing Sciences at Texas A&M University-Corpus Christi. She is interested in multidisciplinary research in imaging applications using a variety of imaging modalities, including thermal imaging, hyperspectral imaging, and other digital imaging that engage targeted sensors, spatial and spectral data processing, pattern recognition and classification. She has a special interest in energy generation and real-world applications, as well as pedagogical methods in teaching and learning.

Hemanth Kumar Reddy Bhimavarapu, Texas A&M University, Kingsville

Hemanth Kumar Reddy is pursuing master's in computer science as well as working on campus as a Graduate Research Assistant in the Electrical Engineering and Computer Science Department at Texas A&M University-Kingsville. Before that he did his Bachelor's in Electronics and Communication Engineering at KL University-Andhra Pradesh, India. He is actively working on developing IOT applications and doing research on U3810A IOT Educational Kit.

Enabling Remote Student Learning of IoT Technologies

Abstract

The proliferation of inexpensive communication, computing, and sensor devices continues to fuel the design and deployment of a wide variety of Internet of Things (IoT) enabled devices. Correspondingly, providing students with educational experiences that include IoT technology has become an increasingly important element of university degree programs. In order to gain practical hands-on experience with IoT applications students require access to the physical devices of which these systems are constructed such as sensors, processor boards, and communication networks. This provides a holistic educational experience in which students can gain proficiency with designing, assembling, and programming IoT based applications.

As classes at many universities have transitioned back to an in-person format, some schools have elected to continue offering at least some of their courses in an online format. Online courses in which students study remotely can pose challenges for students, especially those which are lab-based that require students to have access to physical equipment in order to complete assignments. To support remote student learning, Texas A&M University-Kingsville, a Hispanic Serving Institution, has adopted a strategy that utilizes a basic IoT learning kit. Each kit consists of a basic processor board (such as a Raspberry Pi or a similar board), a variety of sensors, and the necessary equipment to connect components. Students learning remotely that are taking a course involving IoT topics are allowed to check out an IoT learning kit for the duration of the semester. Students can utilize the components of the kit to design solutions for IoT related assignments in their courses.

A series of exercises have been developed at Texas A&M University-Kingsville as part of the capstone senior design course to introduce IoT concepts to remotely located students and help them learn how to use the components of the IoT learning kits. The exercises start with the basics of connecting and reading data from sensors and progress through logging data to a website and then utilizing it to control an IoT enabled device remotely. The IoT learning kits provide the opportunity for remotely learning students to engage with hands-on learning. Thus, students gain a better understanding of IoT concepts and technologies and how they might be integrated into their capstone projects.

Introduction

Problem based learning (PBL) is an area of research that has been shown to increase student interest on engineering topics [1]-[3]. Internet of Things (IoT) enabled devices present an area where PBL methods can be readily utilized. IoT enabled devices continue to become more common in industrial applications, consumer electronics, educational environments and curricula [4]-[7]. IoT example applications can include uses in robotics [6], reading temperature [5] or pressure sensors and health monitoring and applications [8]-[9]. IoT's continued increases in utilization necessitate more educational materials and curricula to include IoT concepts and hands-on practice with IoT applications and devices. This provides a holistic educational experience in which students can gain proficiency with designing, assembling, and programming IoT based applications.

Background

Many courses at Texas A&M University-Kingsville and Texas A&M University-Corpus Christi were affected during the COVID-19 pandemic and were moved online. As a result. most labs had to be modified to allow students to take them remotely and even today many courses still have online sections at Texas A&M University-Kingsville. To support online or off-campus learning, students would be able to check out an IoT kit. The IoT learning kits provide the opportunity for remotely learning students to engage with hands-on learning. Thus, students gain a better understanding of IoT concepts and technologies and how they might be integrated into their capstone projects. The assignments reported in the rest of this paper provide an opportunity for students to learn how to incorporate IoT and are part of IoT related research and materials [9]-[11] that have been developed by the PIs.

IoT Assignments

To introduce students to the use of the Internet of Things (IoT) concept for designs and applications, five assignments have been developed at Texas A&M University-Kingsville to aid students in understanding how to access and control devices over the IoT. In this case a Raspberry Pi board has been utilized as a basis for IoT learning. Other similar FPGA or processor-based boards would also suffice. Later implementations of the assignments will utilize a more advanced board with more capabilities than the Raspberry Pi. Initially the five assignments will be utilized in the Capstone Senior Design courses in Computer Science and Electrical Engineering. Approximately 30 students are enrolled in the two courses during the Fall or Spring semesters.

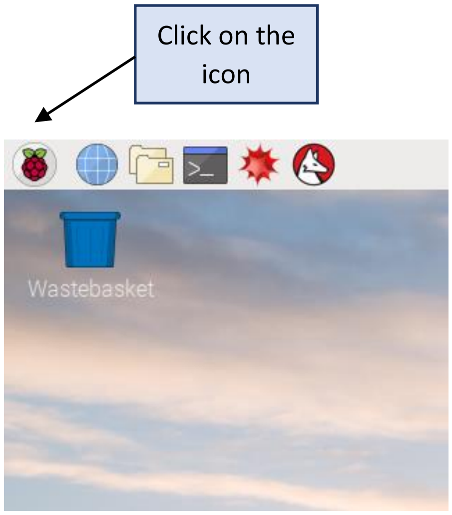
These five assignments build upon the authors' research reported in [10]. These developed IoT materials will enhance the students' IoT skills and confidence to utilize them in further designs such as in their senior design capstone projects. This material is based upon work supported by the National Science Foundation under Grant No. 2044255.

The five assignments include the following topics:

1. Introduction and Running a Simple Program on a Raspberry Pi,
2. Connecting to the Internet with a Raspberry Pi,
3. Getting sensor data readings using a Raspberry Pi,
4. Sending data to the cloud and plotting the data, and
5. Remotely driving a motor, LED or other device over the internet using a Raspberry Pi.

An IoT survey developed for the funded grant will be administered to the students in order to ascertain any knowledge gains concerning IoT.

In assignment 1 [12] the students first learn about connecting the Pi board to a display and opening a programming environment. In the assignment, students will write a short program to output a message such as "Hello World." Python will be used in this assignment but other languages such as C could also be utilized to accomplish this assignment.



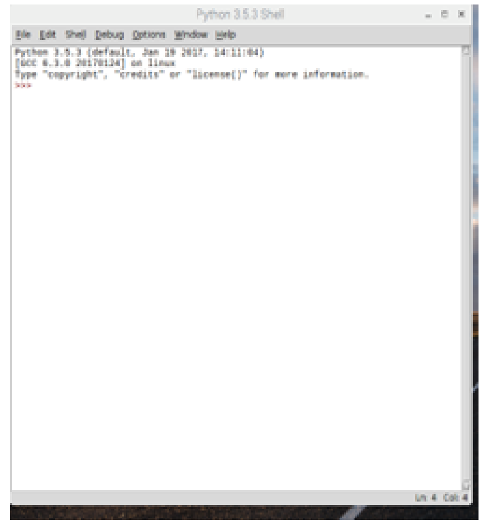


Figure 1 – a) Raspberry Pi Desktop b) Prompt in Python

After the Pi board is connected to a display, a keyboard, and a mouse, the student can power up the Raspberry Pi. The desktop should look similar to Figure 1 a) above. If one clicks on the Raspberry on the display, one can choose a programming language such as Python. This will then bring up a command prompt as seen in Figure 1 b). Example code to print simple messages to the screen is found below in Figure 2. The corresponding output is for the example code is included [12].

Example Code:

```
print("Welcome to the world of Raspberry\n")
print("Let's say Hello to this Environment\n")
print("Just try adding 2 numbers\n")
a=int(input("Enter 1st number "))
b=int(input("Enter 2nd number "))
c=a+b
print("\n the sum of {0} and {1} is {2}".format(a,b,c))
```

Output:

```
Welcome to the world of Raspberry
Let's say Hello to this Environment
Just try adding 2 numbers
Enter 1st number 10
Enter 2nd number 20
The sum of 10 and 20 is 30
```

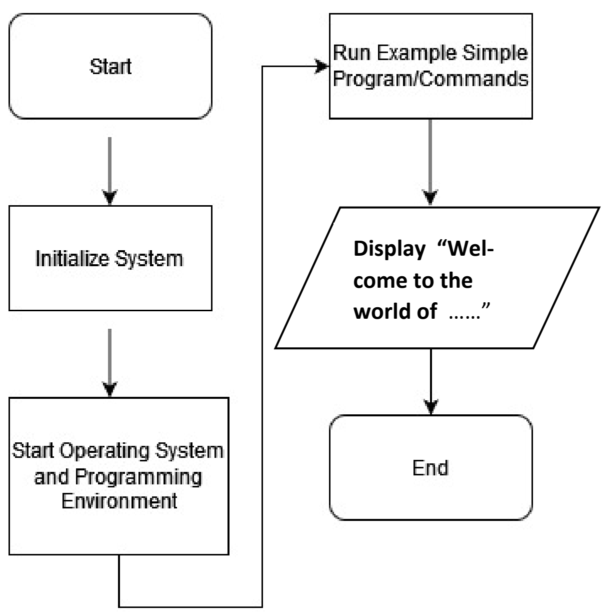


Figure 2 – Example Python Code [12] and Flowchart Modified from [10]

In assignment 2 [13] the student will learn how to connect the Pi to the internet so that in later assignments data and control signals can be transferred over the internet. If a WiFi network is available that is connected to the internet, one can connect to the internet by clicking on the WiFi icon and entering the WiFi login credentials to establish a connection. This is shown in Figure 3 [13]. The user can now check the network interface details by using commands such as "ipconfig" or "ifconfig -a" and to check the connection by using commands such as "ping" from a command prompt as shown in Figure 4 [13].

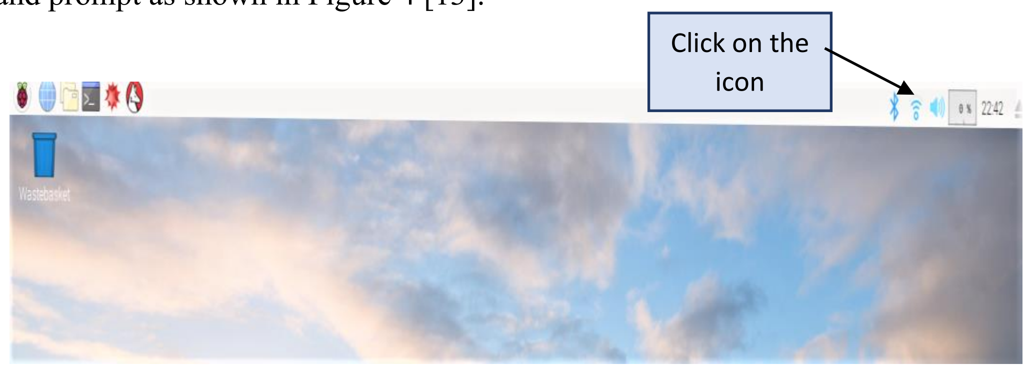


Figure 3 : WiFi icon [13]

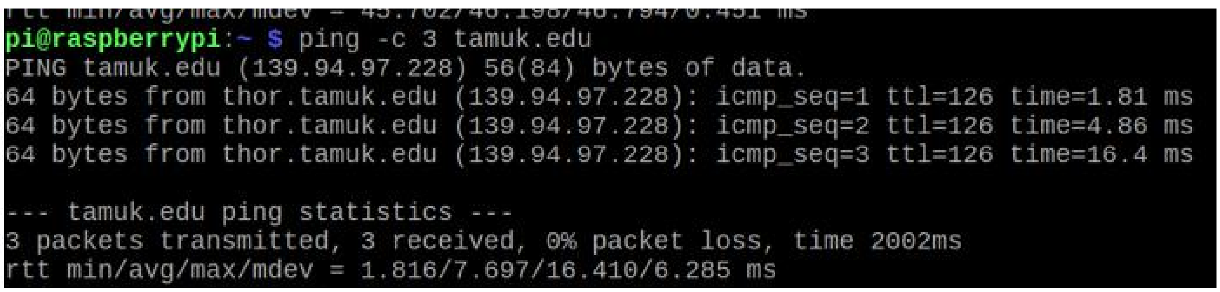


Figure 4 : Ping Command Utilized to Check Internet Connection [13]

The "ping" command will report statistics for sending and receiving packets to/from the website that was "pinged."

In assignment 3 [14] a sensor will be connected to a general-purpose input output (GPIO) pin on the Raspberry Pi. Figure 5 displays a simple circuit for utilizing a simple sensor such as an HW-201 infrared sensor. The output for this sensor is a digital output to simplify the connection to the Pi. Figure 6 displays example code for reading sensor data for the circuit in Figure 5.

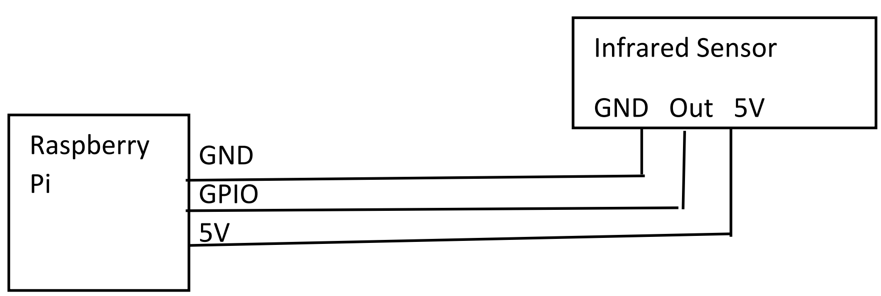


Figure 5 – Simple Infrared Circuit to Determine the Presence of an Object [14]

File Edit Format Run Options Window Help

```python
import RPi.GPIO as IO
import time
IO.setmode(IO.BCM)
IO.setwarnings(False)

input_sensor_1 = 17  # maps to physical pin number 11 on Pi board
IO.setup(input_sensor_1, IO.IN, pull_up_down=IO.PUD_UP)

while True:
    light_or_dark = IO.input(input_sensor_1)

    print("value read from sensor is:  ", light_or_dark)

    if(light_or_dark == 0) :
        print("OBSTACLE DETECTED")
    else :
        print("NO oBSTACLE DETECTED")
    time.sleep(1)

IO.cleanup()
```

Figure 6 –Python Example Code - Read an Infrared Sensor to Determine Presence of an Object [14]

The corresponding output is displayed in Figure 7 [14]. In the code, the output will be “1” or “0”; a “1” denotes no obstacle detected while a “0” represents an obstacle has been detected by the infrared sensor.

File Edit Shell Debug Options Window Help

```
Python 3.5.3 (default, Jan 19 2017, 14:11:04)
[GCC 6.3.0 20170124] on linux
Type "copyright", "credits" or "license()" for more information.
>>>
===================== RESTART: /home/pi/ASSIGNMENT-3.py =====================
value read from sensor is:   1
NO oBSTACLE DETECTED
value read from sensor is:   1
NO oBSTACLE DETECTED
value read from sensor is:   1
NO oBSTACLE DETECTED
value read from sensor is:   1
NO oBSTACLE DETECTED
value read from sensor is:   1
NO oBSTACLE DETECTED
value read from sensor is:   1
NO oBSTACLE DETECTED
value read from sensor is:   0
OBSTACLE DETECTED
value read from sensor is:   0
OBSTACLE DETECTED
value read from sensor is:   0
OBSTACLE DETECTED
```

No Obstacle Detected

Obstacle Detected

Figure 7 –Infrared Sensor Example Code Output [14]

Now that the student has completed Assignments 1-3 the student can connect the Pi to the internet and read sensor data on a GPIO pin. The next Assignment will store data on a webserver. In Assignment 4 one can use a platform such as ThingSpeak [15] to facilitate internet data storage and analytics. One can create a channel in ThingSpeak. With the channel created, one will need to determine the Application Programming Interface (API) key for the channel so one can write to the channel from the Raspberry Pi, see Figures 8 and 9 [16].

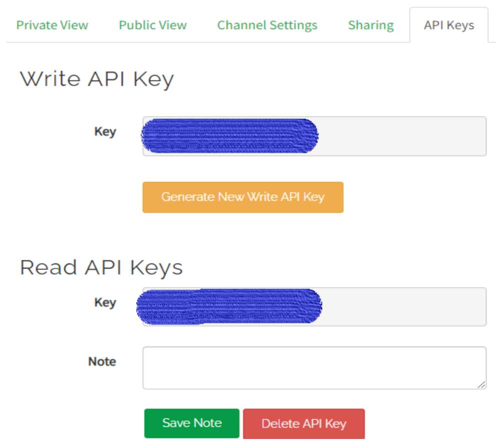


Figure 8 -API Keys for the Channel [16]

```
File  Edit  Format  Run  Options  Window  Help
import httplib
import urllib
import time
import RPi.GPIO as IO
import time
IO.setmode(IO.BCM)
IO.setwarnings(False)
input_sensor_1 = 27  # maps to physical pin number 13 on Pi board
IO.setup(input_sensor_1, IO.IN, pull_up_down=IO.PUD_UP)
key = "FP8K7WTRQP9KLJGW"  # Put your API Key here
def sense():
    while True:

        light_or_dark = IO.input(input_sensor_1)

        print('value read from sensor is:  ', light_or_dark)

        if(light_or_dark == 0) :
         print("OBSTACLE DETECTED")
        else :
         print("NO OBSTACLE DETECTED")
        time.sleep(1)

        params = urllib.urlencode({'field1': light_or_dark, 'key':key })
        headers = {"Content-typZZe": "application/x-www-form-urlencoded","Accept
        conn = httplib.HTTPConnection("api.thingspeak.com:80")
        try:
            conn.request("POST", "/update", params, headers)
            response = conn.getresponse()
            data = response.read()
            conn.close()
        except:
            print("connection failed")
        break
if __name__ == "__main__":
        while True:
                sense()

                                                          Ln: 29  Col: 41
```

Figure 9 – Example Python Code to Send Data to ThingSpeak [16]

Figure 10 shows the overall program flow to post data to a website/cloud server as described in Assignment 4. Assignment 5 [17] will then build upon this to drive a motor or LED using control signals via the internet. This IoT capability could be for example used to control a robot's motors or to send data as part of a user interface.

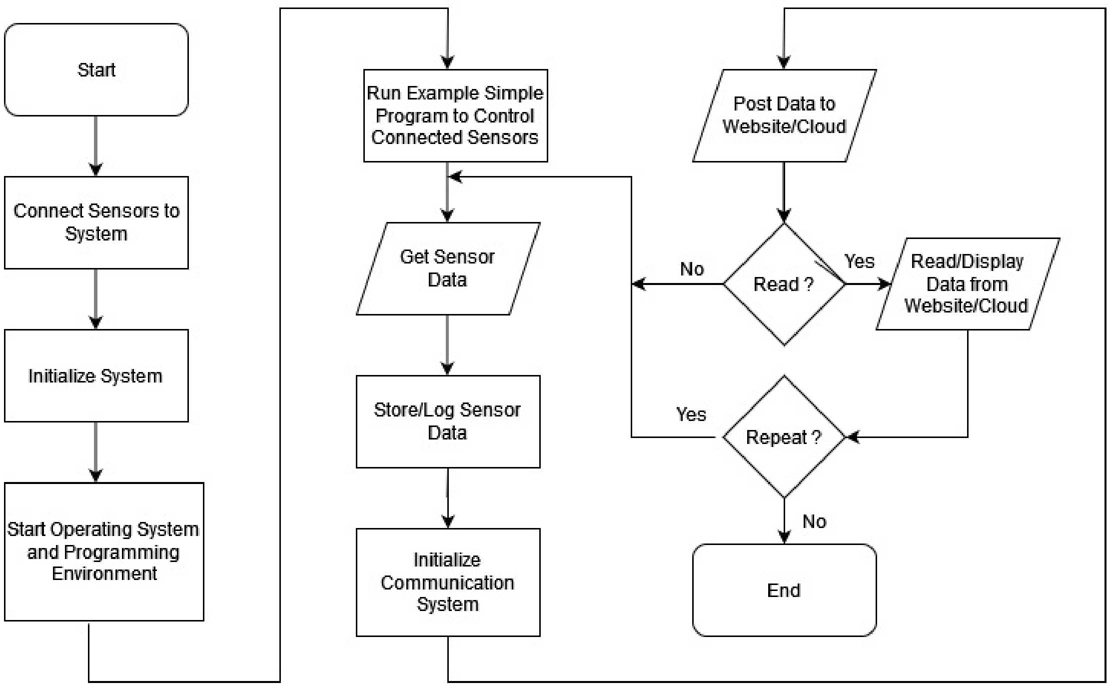


Figure 10 – Flowchart for Assignment 4 – Posting Sensor Data to Cloud [9]

The data on ThingSpeak can be visualized in real-time. Figure 11 [16] displays a plot of the data generated while running the Example Code from Figure 9. Figure 12 shows example Python code to control a motor and an LED [17].

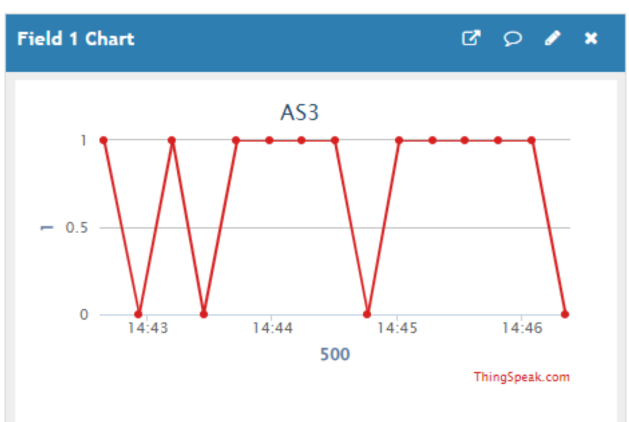


Figure 11 – Output for Example Code on Channel in ThingSpeak [16]

```
import urllib
import urllib2
import time
import RPi.GPIO as IO
import time
import json
Read_Key='Z8UDZRZCX2HD0BWF' # Read Api Key From the Channel
Channel_Id='2033530' #Thingspeak Channel Id
IO.setmode(IO.BCM)
IO.setwarnings(False)
input_sensor_1 =  27  # maps to physical pin number 13 on Pi board
output_sensor_1 = 6   # maps to physical pin number  31 on Pi board
IO.setup(input_sensor_1, IO.IN, pull_up_down=IO.PUD_UP)
IO.setup(output_sensor_1,IO.OUT)
IO.setup(7,IO.OUT) #Motor 1 assignment to GPIO 7
IO.setup(8,IO.OUT) #Motor 1 assignment to GPIO 8
p=IO.PWM(7,50)
q=IO.PWM(8,50)
p.start(0)
q.start(0)
v=20                   #desired speed of the wheel
key = "FP8K7WTRQP9KLJGW"  # Put your API Key here
def sense():
    while True:

        light_or_dark = IO.input(input_sensor_1)
        print('value read from sensor is: ', light_or_dark)
        time.sleep(1)

        params = urllib.urlencode({'field1': light_or_dark, 'key':key })
        headers = {"Content-typZZe": "application/x-www-form-urlencoded","Accept": "text/plain"}
        conn = httplib.HTTPConnection("api.thingspeak.com:80")
        ts=urllib.urlopen("https://api.thingspeak.com/channels/2033530/feeds.json?api_key='Z8UDZRZCX2HD0BWF'&result
        try:
            conn.request("POST", "/update", params, headers)
            response=ts.read()
            data=json.loads(response.decode('utf-8'))
            #print(data)
            print("Value Read from channel ",data["feeds"][1]["field1"]) #This the value read from our cloud channe
            print('\n')
            a=data["feeds"][1]["field1"]
            if a=='0':
                IO.output(output_sensor_1,1) # setting red led to HIGH When there is an object detected
                p.ChangeDutyCycle(0)         # Turn off the motor when object detected

            elif a=='1':
                IO.output(output_sensor_1,0) # setting red led to LOW
                p.ChangeDutyCycle(v)         # Motor will be onn when there is no object

            conn.close()

        except:
            print("connection failed")
        break
if __name__ == "__main__":
        while True:
                sense()
```

Figure 12 – Example Python Code to Control a Motor and an LED [17]

In Figure 13, the Raspberry Pi, infrared sensor, motor and an H-bridge motor driver are pictured. The H-bridge motor driver is utilized in this case to simplify the circuit to have the Raspberry Pi control the motor. The motor speed can be adjusted inside the Python code.

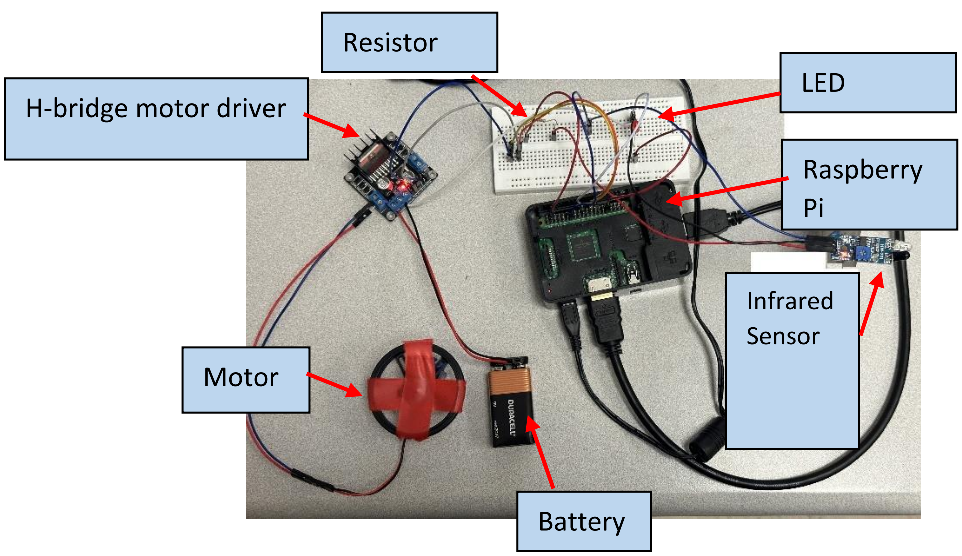


Figure 13 – Setup with Motor [17]

Conclusions

Initially the five assignments will be utilized in the Capstone Senior Design courses in Computer Science and Electrical Engineering. Students will gain experience working with IoT applications through the Assignments 1-5. The IoT learning kits provide the opportunity for remotely learning and in-person students to engage with hands-on learning. Thus, students gain a better understanding of IoT concepts and technologies and how they might be integrated into their capstone projects. Later implementations for these and future assignments will utilize a more advanced board with more capabilities than the Raspberry Pi. An IoT survey developed for the funded grant will be administered to the students in order to ascertain any knowledge gains concerning IoT after they complete the five assignments.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 2044255.

References

[1] Michael Prince, "Does Active Learning Work? A Review of the Research," *Journal of Engineering Education*, pp. 223-231, July 2005.
[2] A. M. Rad, T. H. Popa, V. -D. Mihon and B. Iancu, "Problem-based learning and project-based learning concepts and their applications to engineering education," *2017 16th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, pp. 1-6, 2017.
[3] L. McLauchlan and M. Mehrubeoglu, "A Laboratory Exercise - Unmanned Vehicle Control and Wireless Sensor Networks," *2014 ASEE Annual Conference and Exposition,* Indianapolis, IN, USA June 15-18, 2014.
[4] J. Agrawal, O. Farook, Z. Anderson and D. Walker, "Internet of Things (IoT) Laboratory," *2019 ASEE Annual Conference and Exposition,* Minneapolis, Tampa, FL, USA June 15-19, 2019.
[5] V. Chang and C. Martin, "An industrial IoT sensor system for high-temperature measurement," *Computers and Electrical Engineering*, (95), pp. 1-13, 2021.
[6] J. Jones, "A Hands-on Introduction to Embedded Systems & IoT," *ASEE's Virtual Conference,* June 22-26, 2020.
[7] G. Mullett, "Teaching the Internet of Things (IoT) Using Universally Available Raspberry Pis and Arduino Platforms," *2016 ASEE Annual Conference and Exposition,* New Orleans, LA, USA June 26-29, 2016.
[8] K. Sangeethalakshmi, S. Preethi Angel, U. Preethi, S. Pavithra, and V. Shanmuga, "Patient health monitoring system using IoT," *Materials Today: Proceedings,* Available online June 24, 2021.
[9] A. Moore, M. Mehrubeoglu, A. Mooney, and L. McLauchlan, "Application of IoT-based Sensing and Signal Processing for Rehabilitation," *Proc. of SPIE*, April 2022.
[10] L. McLauchlan, D. Hicks, M. Mehrubeoglu, and B. Zimmer, "Work-in-Progress: Internet of Things Enabling Remote Student Learning," *2022 ASEE Annual Conference and Exposition,* Minneapolis, MN, USA June 26-29, 2022.
[11] M. Mehrubeoglu, L. McLauchlan, D. Hicks, and B. Zimmer, "Work-in-Progress: Developing an IoT-based Engaged Student Learning Environment and Tools for Engineering

Programs," *2022 ASEE Annual Conference and Exposition,* Minneapolis, MN, USA June 26-29, 2022.
[12] H. Bhimavarapu, L. McLauchlan, and D. Hicks, IoT Assignment 1, Department of Electrical Engineering and Computer Science, Texas A&M University-Kingsville, 2023.
[13] H. Bhimavarapu, L. McLauchlan, and D. Hicks, IoT Assignment 2, Department of Electrical Engineering and Computer Science, Texas A&M University-Kingsville, 2023.
[14] H. Bhimavarapu, L. McLauchlan, and D. Hicks, IoT Assignment 3, Department of Electrical Engineering and Computer Science, Texas A&M University-Kingsville, 2023.
[15] IoT Analytics - ThingSpeak Internet of Things, https://thingspeak.com/, last accessed February 28, 2023.
[16] H. Bhimavarapu, L. McLauchlan, and D. Hicks, IoT Assignment 4, Department of Electrical Engineering and Computer Science, Texas A&M University-Kingsville, 2023.
[17] H. Bhimavarapu, L. McLauchlan, and D. Hicks, IoT Assignment 5, Department of Electrical Engineering and Computer Science, Texas A&M University-Kingsville, 2023.