Considering the design and operation of a voice controlled laboratory for teaching and learning in electronic engineering

Ian Grout
Department of Electronic and Computer Engineering
University of Limerick
Limerick, Ireland
e-mail: Ian.Grout@ul.ie

***Abstract*—In this paper, the design and operation of a voice controlled electronic engineering laboratory for the teaching and learning of semiconductor device current-voltage (IV) characteristics are presented. This is targeted to students studying electronic engineering in a higher education setting. The presented case study experiment operation, based on a Schottky diode forward and reverse bias IV characteristics, is presented. This uses a computer-based interface to an Arduino UNO platform that acts as the experiment test and measurement equipment. The computer side software developed is based on a Python script along with a speech recognition library and the online Google Speech Recognition service for identifying voice commands**.

Keywords—Laboratory experiments, speech recognition, voice control, accessibility

I. Introduction

When considering the design and operation of laboratory experiments in the teaching and learning of electronic engineering concepts, it is common to use some form of computer-based test and measurement equipment arrangement. The typical approach for interacting with the equipment is through a software application running on the computer that sends commands to the equipment and receives experimental data from the equipment. The user interactions are usually designed for a *typical* student who would use a keyboard and mouse arrangement to input the required commands and uses a visual display unit (VDU) to output system status information and experimental results data. This *typical* is based on the idea of a *one size fits all* approach. However, the keyboard and mouse input approach, whilst suitable for many students, is not necessarily supportive for all students, for example those with a disability. The keyboard and mouse *only* interface might not be inclusive for all students.

Providing for alternative input approaches that allow the student to select the most appropriate technology. or suite of technologies to use, would provide the laboratory and experiment to be accessible to more students. This would follow a Universal Design for Learning (UDL) [1] approach. UDL is "*a set of principles for curriculum development that give all individuals equal opportunities to learn*" [2]. UDL provides a framework to improve the learning experience of all students within the mainstream teaching environment irrespective of the individual student circumstances. UDL is based on three core principles to provide multiple means of *engagement*, *representation*, and *action & expression*.

In this work, the principle of providing *multiple means of action & expression* is considered within the laboratory environment. Here, the keyboard and mouse input devices are augmented by a microphone input to provide voice control of the experiment operation using speech recognition. The principle is applied to a case study experiment design, although in general the principles can be applied to multiple experiments by customising the approach adopted to a particular experiment. Speech recognition [3] is the ability for a machine or program to identify spoken words and to convert them into readable text. Speech recognition is also referred to as speech-to-text [4]. A user speaks into a microphone, and the speech recognition program interprets the audio signals received to convert the audio signals into understandable words in a selected language. Speech recognition is used in this work within a laboratory setting to control the operation of an electronic circuit experiment. The program running on the computer is written to use a specific speech recognition API (application programming interface) that implements the speech-to-text operation. The experiment will be running within a laboratory setting, and the laboratory can be either an at-presence (face-to-face) laboratory or a remote (online) laboratory. In general, the different forms (classes) of laboratory are identified in Table I. Remote laboratories support remote, or virtual, learners [5].

TABLE I. Classification of Laboratories

	Local experimenter (same location as the experiment)	Remote experimenter (different location from the experiment)	
Experiment type	Local	Remote	
Real (physical)	At-presence (physical)	Remote laboratory	Hybrid
Virtual (simulated)	At-presence (simulated)	Virtual laboratory	

The classification here is based on whether the experimenter is local to the experiment (experimenter is physically in the same laboratory as the experiment – "*at-presence*") or remote (different location and access is via an internet connection – "*remote*"), and whether the experiment is a real physical experiment or a software simulation of the real physical experiment.

The experiment presented in this work is based on the design of the laboratory set-up using the Arduino UNO [6] platform and the Python [7] programming language. The SpeechRecogntion [8] library for Python and the Google Speech Recognition API [9] are also utilised.

This paper will be presented as follows. Section II will discuss means of access in undertaking laboratory experiments using computer-based test and measurement equipment. Section III will present the Schottky diode experiment laboratory hardware based on an Arduino UNO platform. Section IV will present the software interface based on a Python script. Section V will discuss the system operation, and section VI will provide conclusions.

II. MEANS OF ACCESS TO LABORATORY EXPERIMENTS

In order to access a laboratory experiment, the way in which the student proceeds through the complete process from preparation to submission of completed assessment materials need to be considered. Analysis of such procedures would show that the laboratory is a dynamic environment, with different requirements, and different identifiable outcomes. The procedure would be established by the education provider based on a number of criteria, for example the capabilities of the provider to ensure that the right learning outcomes are attained. The viewpoint of the provision can be either provider-centric or student-centric, that is considering who is of the primary concern. Ideally, the laboratory would be set-up with the student at the centre of the decision-making process. If a Universal Design for Learning (UDL) [1] approach is adopted, the student would be of primary concern.

The three principles of UDL could be applied to the different aspects of the laboratory teaching and learning approach to be adopted. If the third principle, to *provide multiple means of action & expression* was to be considered, this aims to provide options for *physical action*, *expression & communications*, and *executive functions*.

Considering these three categories of options, the manner in which the student would interact with the laboratory equipment, the experiment, and teaching staff, could be reviewed and analysed with the ultimate aim to provide options that would best suit the individual student whilst ensuring equity to all students, meet the required quality levels, and ensure that the required learning outcomes are met. In this paper, the role of the computer input device option is considered. This can be seen with reference to Fig. 1. Specifically, a laboratory that allows the laboratory user to interact with different HCI (human-computer interaction) [10] devices would support the ability to *provide options for physical action.* The *typical* way in which the user interacts with the laboratory computer is via a keyboard and mouse combination. This is usually the only approach that would be considered. However, for individuals who may not be able to access the keyboard and mouse, alternative devices such as head/hand motion/gesture, eye tracking, and speech recognition form alternative approaches for interaction.

These different technologies are today seen in various scenarios that can also be used to support the education process. The role of speech recognition is found today in a range of systems such as voice, digital and virtual *assistants* which include Siri/HomePod (Apple), Alexa/Echo (Amazon), Google Assistant/Google Home (Google), and Cortana (Microsoft). These use artificial intelligence (AI) to recognize and respond to voice commands, and speech recognition can be linked to voice control of smart devices for applications such as home automation. Speech recognition has also been applied to running experiments, for example the Voice Experiments Showcase from Google [11].

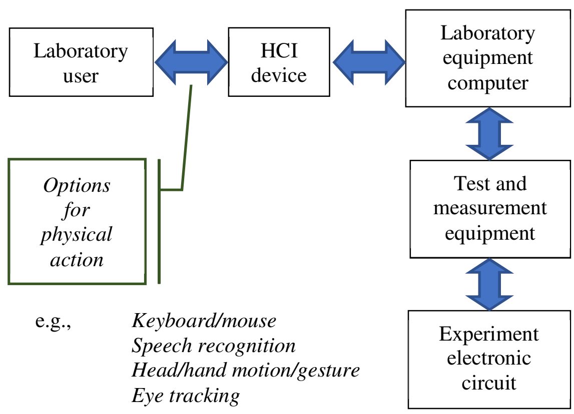

Fig. 1. Electronic circuit experiment access via a HCI device.

III. LABORATORY EXPERIMENT HARDWARE

The experiment to run is based on determining the current-voltage (IV) characteristic(s) of semiconductor device(s), and would be based on one of a number of semiconductor device experiments within a suite of experiments for students to access. In this case study, a BAT86 Schottky diode [12] is to be investigated as an important semiconductor component that students are introduced to when studying electronic and computer engineering. Both forward bias, see Fig. 2, and reverse bias characteristics are to be determined using the experiment hardware.

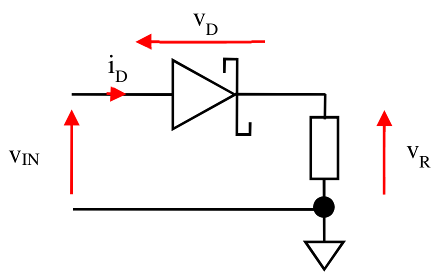

Fig. 2. Diode experiment circuit – forward bias.

The reverse bias characteristic in this case can be determined by reversing the connections of the diode in the experiment circuit. The diode current is determined by measuring the resistor voltage, and calculating the current using the resistor value. In general, multiple analog voltage outputs and analog voltage inputs can be incorporated into the experiment hardware for more complicated circuits. The limit to the number of analog input/output (I/O) channels would be based on the capabilities of the microcontroller to be used in terms of available pins for connecting external peripherals, and the speeds required for analog signal creation and analog signal capture. Fig. 3 shows the hardware set-up based on the Arduino UNO platform used here. The analog voltage to apply to the experiment circuit is created by a 12-bit digital-to-analog converter (DAC). The microcontroller analog inputs using the internal 10-bit analog-to-digital converter (ADC) sample the analog voltages in the experiment circuit. An accurate 2.7 V reference voltage (V_{REF}) is generated for the DAC. An OLED (organic light emitting diode) and two LEDs are included for a system status display. Hence, the analog signal generator uses a 12-bit resolution DAC, and the analog signal sampler uses a 10-bit resolution ADC.

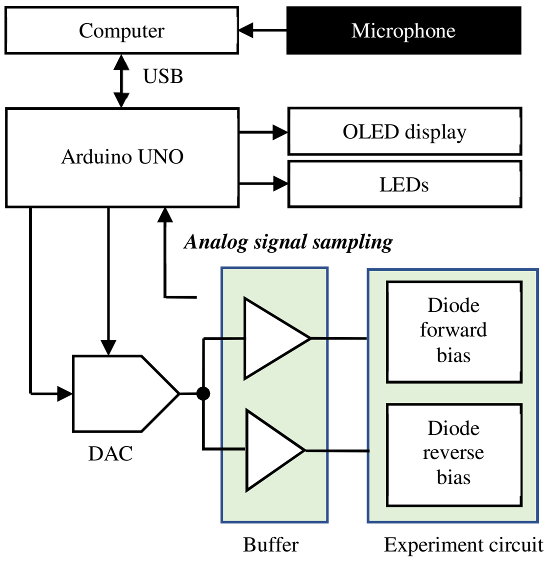

Fig. 3. Diode experiment set-up.

A photograph of the hardware is shown in Fig. 4. Microcontroller programming and system command/data transfer is via the same USB (universal serial bus) cable. The experiment consisting of two diodes and two resistors can be seen to the top right of the prototyping board.

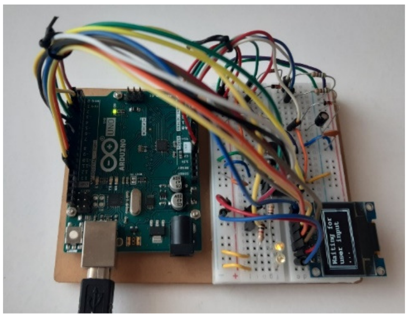

Fig. 4. Diode experiment hardware.

IV. Laboratory Software

The overall system consists of the hardware and software parts. The software itself is based on two parts:

1. The microcontroller program providing the interface between the computer and the experiment. This is based on using the Arduino IDE (integrated development environment) [13]. Alternatively, the microcontroller could be programmed directly in C/assembler using Atmel Studio 7 [14].

2. The computer-side program creating the user interface, speech recognition, Arduino UNO control signals (output), experimental data (input), and experiment results analysis using Python.

The computer-side program is based on a Python script that uses the SpeechRecogntion [8] library for Python and the Google Speech Recognition API [9]. This requires an internet connection as the Google Speech Recognition API is an online service. When the Python script is run, it listens for an audio input from the user. When it detects an input, it attempts to recognize the words spoken. If it detects specific words or phrases, a command is sent to the microcontroller which runs the experiment or initiates an internal operation. The experiment results are automatically returned to the Python script and stored in arrays where the user can then plot and save the results.

V. Laboratory Operation

This section discusses the laboratory operation and shows the results of an example experiment run. Once the experiment has started, the user is prompted to say a word or phrase that is understood in order to run a specific experiment operation. If the word or phrase is understood, the Python script would either send an appropriate command to the Arduino UNO which then runs the particular command and sends results back to the Python script or, run an internal command such as plot results or end (**quit**) the experiment run. The words and phrases used for controlling the experiment, along with plotting and saving the experiment results, are shown in Table II. Once the Python script has been started, control of the script and hence the operation of the experiment would be via voice control only. Hence, this work considers the experiment interactions once the required hardware and software has been set-up.

TABLE II. Voice Commands Summary

Voice Command	Description
Forward bias	Select the diode in forward bias.
Reverse bias	Select the diode in reverse bias.
Input x.y	Apply the experiment input voltage *x.y* V.
Equation x.y	Calculate the diode forward bias equation from 0 V to *x.y* V.
Show results	Show the results obtained as data arrays.
Plot forward	Plot the forward bias experiment results collected.
Plot equation	Plot the calculated forward bias experiment results collected.
Plot forward both	Plot both the diode forward bias experiment results and the calculated diode forward bias equation.
Plot reverse	Plot the reverse bias experiment results collected.
Plot both	Plot both the forward and reverse bias IC characteristics using the acquired results.
Save	Save the acquired experiment results to a results CSV (comma separated values) file.
New	Reset the experiment and clear the results arrays.
Help	Display the help menu.
Quit	Quit the application to end the experiment.

There is a need to use suitable words and phrases that would be meaningful to the context of the experiment and which would not easily be misinterpreted by speech recognizer software. There is also a need to gracefully deal with incorrectly understood words or phrases. For example, the phrase "*Set 2.5*" could have been used as the command to set the experiment input voltage to 2.5 V and read the resulting voltages. However, it is possible for the speech recognizer to misinterpret this to "*Set to .5*". This possibility could have been explicitly dealt with in the code or, as here, this possibility was removed by using the phrase "*Input 2.5*". For demonstration purposes, the Python script was run from the Windows operating system Command Prompt. A sample snippet of the text-based user display is shown below. However, the user interface (UI) can be determined to also best meet the user requirements. Here, the experiment is started, diode forward bias is selected, and an input voltage of 1.5 V selected before quitting the experiment run.

```
-------------------------------------------------------
BAT86 Schottky diode Arduino voice controlled experiment
-------------------------------------------------------
forward
Hello
Let me know what you would like to do.
forward bias
Let me know what you would like to do.
set 1.5
Set(1.5)

--1.5--
-- Start
i = 0  j = 1164
Vref (calibrated) = 2.640 V
Caclulated LSB = 0.004853 V
Vref from LSB = 544 bit code = 2.640 V
-- Forward bias
Vin (forward) = 311 bit code = 1.509 V
Vout (forward) = 250 bit code = 1.213 V
Vd (forward) = 0.296 V
-- End
Let me know what you would like to do.
quit
Goodbye
```

Fig. 5 shows an example plot showing the diode forward bias IV characteristic (I_D versus V_D) where in input voltage (V_{IN}) was varied from 0.0 V to 4.5 V in 0.5 V steps.

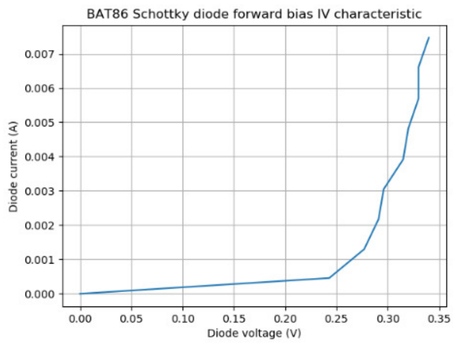

Fig. 5. Example results from an experiment run: forward bias.

A smaller input voltage step size would result in a smoother curve, but the shape of the forward bias IV characteristic can be seen with an exponential rise in the diode current with a rise in the diode forward voltage. As well as a text based visual interface with different wording styles or languages, text-to-speech output as well as a graphical user interface (GUI) based approach could be adopted to give the user access to *multiple means of action & expression*. Hence, it is the ability to provide multiple ways to access the experiment which is of importance. Ultimately however, it is important to consider the complete lifecycle of the experiment from preparation through to the submission of the completed laboratory assessment material. That is, the complete laboratory experiment is considered for accessibility rather than just one aspect.

VI. Conclusions

This paper has presented and discussed the design and operation of voice-controlled experiments for use in the teaching and learning of electronic engineering concepts. A case study laboratory for testing the forward and reverse bias IV characteristics of a Schottky diode was presented with voice control using Python and a speech recognition library. The experiment hardware was based on the Arduino UNO platform. The experiment was operated using voice commands to obtain, plot, and save the experiment results.

References

[1] CAST, About Universal Design for Learning. Accessed on: November 1, 2020. [Online]. Available: https://www.cast.org/impact/universal-design-for-learning-udl

[2] AHEAD, Universal Design for Learning. Accessed on: November 1, 2020. [Online]. Available: https://www.ahead.ie/udl

[3] C. Pearl, Designing Voice User Interfaces: Principle of Conversational Experiences, Chapter 4 "Speech Recognition Technology", O'Reilly, United States of America, 2016, ISBN-13 : 978-1491955413

[4] S. Furui, T. Kikuchi, Y. Shinnaka and C. Hori, Speech-to-text and speech-to-speech summarization of spontaneous speech, IEEE Transactions on Speech and Audio Processing, Vol. 12, Issue 4, July 2004, pp. 401-408,

[5] R.M. Palloff and K. Pratt, The Virtual Student, A Profile and Guide to Working with Online Learners, Jossey-Bass, United States of America, 2003, ISBN 0-7879-6474-3

[6] Arduino UNO. Accessed on: November 1, 2020. [Online]. Available: https://store.arduino.cc/arduino-uno-rev3

[7] Python. Accessed on: November 1, 2020. [Online]. Available: https://www.python.org/

[8] SpeechRecognition library. Accessed on: November 1, 2020. [Online]. Available: https://pypi.org/project/SpeechRecognition/

[9] Google Speech Recognition API. Accessed on: November 1, 2020. [Online]. Available: https://cloud.google.com/speech-to-text

[10] Microsoft, Being Human - Human-Computer Interaction in the Year 2020. Accessed on: November 1, 2020. Available: http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=2309C951C29857911207F9A2EBCC2ED9?doi=10.1.1.153.4252&rep=rep1&type=pdf

[11] Google, Voice Experiments Collection. Accessed on: November 1, 2020. [Online]. Available: https://experiments.withgoogle.com/collection/voice

[12] Nexperia, BAT86 Schottky barrier single diode. Accessed on: November 1, 2020. [Online]. Available: https://www.nexperia.com/products/diodes/schottky-diodes-and-rectifiers/schottky-diodes-and-rectifiers-if-lt-1-a/BAT86.html

[13] Arduino, Downloads, Software IDE 1.8.13, Download. Accessed on: November 1, 2020. [Online]. Available: https://www.arduino.cc/en/software

[14] Microchip, Atmel Studio 7. Accessed on: November 1, 2020. [Online]. Available: https://www.microchip.com/mplab/avr-support/atmel-studio-7