Journal of Research in Engineering and Computer Sciences
April 2024, Vol. 2, No. 2, pp. 23-33

Novel Prototyping Development Board (Source-Era) for an Encompassing Software/Hardware Production

Abubakar Ibrahim Musa[1], Mahmoud Abdulhamid[2], Bappah Adamu Umar[1], Izuchukwu Francis Okafor[3] and E. C. Anne[2]

1. National Centre for Petroleum Research and Development, Abubakar Tafawa Balewa University, Bauchi State, Nigeria
2. Department of Electrical and Electronics Engineering, Abubakar Tafawa Balewa University, Bauchi State, Nigeria
3. National Centre for Energy Research and Development, University of Nigeria, Nsukka, Enugu State, Nigeria

Abstract:
In programming and reprogramming of a controller, a setup known as Prototyping platform is required, and three things are involved: Software IDE to write the code, hardware development board to transfer the code and the prototype components to assemble and test the work. Prototyping platforms give software/hardware developers a platform to ease the design and implementation of complete electronic solutions that lead to quality products. To encourage an increase in numbers of developers and thus products, such platforms must meet the requirements of ease of use, expandability and cost which most existing platforms lack. The need for cost effective development board that is robust, expandable and ease of use informed this work. The introduction of Novel Prototyping Development Board (Source-Era), is a big boost to software/hardware production. The board explores microcontrollers such as PIC18F873, PIC16F876, dsPIC, ATMEGA328A and more without changing the development board or performing any soldering activities. It also has the ability of interfacing a controller with different input and output device based on their signal type which include Universal Asynchronous Receive and Transmit (UART) Protocol, Serial Peripheral Interfacing (SPI) Protocol, Inter Integrated Chip (IIC or I2C) protocol, Analog signal processing protocol (ASP), Boolean Digital, Pulse width Modulation (PWM).

Keywords: Programming, Microcontroller, Prototyping, Platform, Development, Board, Expandable, Cost.

INTRODUCTION

Electronics and software engineering are one of fast-growing fields today. However, beginners /learners are often entangled with an abstract system of learning, which makes their level of understanding quite inadequate and most often find it difficult to easily fit into their professions. In the course of their learning as beginners, they are subjected to build devices they barely know the concept and effects of most components; thus, this deters their innovativeness. However, beginners/learners are enthusiastic about seeing an embedded system which is designed, programmed and developed by themselves, working effectively for the purpose of which the embedded systems were made for. The embedded system is in general, a special-purpose computer system, all of which are put into a controlled device. It has certain needs and performs pre-set tasks, unlike a multipurpose personal computer. Examples of systems or applications include medical instrumentation (for example, smart thermometer), process control, automated vehicles control, and communication devices. Modern embedded systems are often based on microcontrollers (i.e., microprocessors with integrated memory and peripheral interfaces) such as Arduino Mega and Raspberry Pi. A general-purpose definition of Embedded System is that they are devices used to control, monitor, or assist the operation of equipment, machinery, or other big systems (Abubakar and Mashoedah, 2021).

The invention and development of microcontrollers relates closely to the evolution of embedded systems while developments in the areas of embedded system ignited the emergence of new platforms. These platforms are based on computer and embedded system to control and monitor all the matters without human's direct involvement. Embedded systems have become an integral part of daily life. Embedded systems like a cell phone, a smartcard, a music player, a router, or the electronics in an automobile have been touching and changing modern lives like never before. An embedded system is a combination of computer hardware, software, and additional mechanical or other technical components, designed to perform a dedicated function. Most of the embedded systems need to meet real time computing requirements. The major building blocks of an embedded system are listed below:

i. Microcontrollers / digital signal processors (DSP)
ii. Integrated chips.
iii. Real time operating system (RTOS) - including board support package and device drivers.
iv. Industry-specific protocols and interfaces.
v. Printed circuit board assembly

Usually, an embedded system requires mechanical assembly to accommodate all the above components and create a product or a complete embedded device (Patil and Tupe, 2020). The usual alternative to building application-specific circuits is to use a general-purpose processor, and customize it for an application by writing a program, thus, enlarging the coast of innovation. But for some applications, particularly in need of speed for execution or security is important, a custom-built circuit has some advantages over the usual processor-and-software combination. Therefore, a prototyping platform will give software/hardware developers an avenue to ease the design and implementation of complete electronic solutions that may lead to quality products. In order to encourage an increase in numbers of developers and products, such platforms must meet the requirements of ease of use, expandability and cost. The objective of crafting this prototyping board is to offer a platform that is cost effective, easy to use and expandable.

LITERATURE REVIEW

Microcontroller is a mini computer system on a single integrated circuit. Contemporarily, it is a System on a chip. A microcontroller contains one or more central processing units (CPUs) along with memory and programmable Input Output peripherals. Microcontrollers are generally classified according to processor characteristics such as processor architecture, processor word processing length, processor clock operating frequency, and so on. While reduce instruction set computer (RISC) architectural microprocessors are generally preferred in electronic applications such as control and automation, complex instruction set computer (CISC) architectures are used in microcomputer applications such as signal processing and embedded systems. There are some important features that distinguish microcontrollers from each other which include; Processor Architecture, type of application and command set used, Processor word length, length and type of data to be processed, Processor clock frequency and the processing speed of the codes to be executed. The random-access memory (RAM) and read only memory (ROM) has the capacities of accommodating the program and data, with Input/Output units, and data exchange, and communication with external media. A well-designed microcontroller has a special built-in enhancement, form factor, Physical structure and working environment. Operating conditions, Supply voltage, electromagnetic compatibility, compatibility with other circuits (Yilmaz, 2017) are critical need that cannot be traded off. There are two types of Microcontrollers; one is Embedded Microcontroller and the other is External Memory Microcontroller. External memory microcontroller is used when extra memory is needed, therefore need can fulfilled by allowing the connection of external memory during the work as a separate ROM (RAM) which will make the work simpler and less sophisticated (Hollenthoner, 2007)). An embedded microcontroller is a chip which has a computer processor with all its support function, memory, and input-output interfacing built into the device. In teaching and prototyping, there are many different microcontroller development tools available for use. Those that are most popular outside the electrical engineering community work to offer some balance between cost, expandability, and ease of use. Arduino also seeks to balance these factors, while making up for some of the shortcomings of existing platforms. The Microcontroller technology encompasses the software requirements and the hardware development platforms.

The technology development in programming languages is rapid. Simply, programming as a problem has only arisen since computer machines were first created. The magnitude of the problem is however relative to the size (and complexity) of the computer machine used (Cook, 1999). To program a computer system, a programming language is required. A programming language should be reasonably an instruction for solving problems, at least problems within its intended application area. For example, a programming language whose only data types are numbers and arrays might be natural for solving numerical problems, but would be less natural for solving problems in commerce or artificial intelligence. Conversely, a programming language whose only data types are strings and lists would be an unnatural choice for solving numerical problems (Chen, 2010). In Mobile Computing, Applications Software relies on the operating systems for most I/O tasks. Different operating systems uses different methods to access hardware, making application coded in languages like C++ unable to run under other operating systems than what they were written for. One way to solve this problem is to introduce a software layer between the application and the operating system. Learning the principles of how the Input/Output (I/O) subsystem of a computer functions is important for students taking computing degrees, not only because a computer engineer or scientist should not regard a computer as just a black box, but also because the management of the I/O subsystem can greatly affect the performance of a machine and is a major component of most embedded systems (Edurne, 2013). Microcontroller hardware platforms are available in the market and consist of the following: 1). Microdig 2). Phidgets 3). D.tool 4). Programma 2003 and 5). Arduino Family alongside NI myRIO Boards and Raspberry Pi Board

Reas (2005) reports that MicroDig, Phidgets, and Stanford's tools are modules that are generally not programmable by the end user. Instead, they are configured using a desktop tool. These tools are generally not standalone devices, but must be connected to a personal computer in order to be useful. MicroDig is a sensor interface box with a musical instrument digital interface (MIDI). The hardware interface consists of an analog to musical instrument digital interface, (Analog-to MIDI) controller with 8 analog inputs, and various sensor modules that mate with the controller. Users attach pre-packaged sensors to the inputs, and connect the controller to a musical instrument digital interface MIDI output device. The values of the sensors are output as musical instrument digital interface, MIDI values. The MicroDig is handy for teaching students with some knowledge of MIDI but little programming or electronics knowledge on how to design hardware interfaces, because it requires little new knowledge. It is an expensive platform and requires that the connecting equipment be MIDI compatible (Philips, 2009). Phidgets is a modular system of sensor controllers, motor controllers, radio frequency identification (RFID) readers, and other special function devices, all united by a common universal serial bus (USB) interface and a set of desktop software application programming interface (API). Each Phidget device is a self-contained electronic device, whether it is a sensor, motor or light emitting diode (LED) controller, or a more complex device like an LCD display. The user needs almost no knowledge of electronics to use Phidgets. Each device is connected to a desktop computer in order to access its sensor data or to control it. The development team has released application programming interfaces for the system in several languages, including Visual Basic, visual basic analysis (VBA), (Microsoft Access and Excel), LabView, Java, Delphi, C, C++, and Max/MSP. The modules are relatively affordable (Li Xi, 2005). The devices cannot be used as standalone units, but must be interfaced to a personal computer to use. The learning curve for Phidgets is somewhat steeper than for the MicroDig, but it's useful for those familiar with software development who want to begin making hardware interfaces (Phipps, 1999). D-tools is a high-level hardware and software tool developed at Stanford University's HCI group that addresses some of the shortcomings of others in this class. First, D-tools is a more flexible system. The D-tools software can be used with other hardware platforms, as long as that hardware is running a firmware that can communicate in the D-tools protocol. Wiring Arduino and Phidgets hardware have been used with D-tools. The software is written in Java as a plugin for the Eclipse universal tool platform, and can theoretically run on any Java-capable operating system. Like Phidgets, the hardware is made up of a series of plug-and-play USB modules, each of which communicates with the tool's software. D-tools also offers a suite of analysis tools which allow users to see the results of their devices graphed on screen, and time-indexed against a video of the person using the device.

Arduino is an open-source microcontroller that can easily be programmed, erased and reprogrammed at any time as desired. Arduino is a small microcontroller board with a universal serial bus (USB) plug that can be connected to a computer and a number of connecting sockets that can be wired to external electronics such as motors, relays, light sensors, laser diodes, loudspeakers, microphones, and a lot more desired electronic device. They can either be powered through the USB connection from the computer, 9V battery, or from a power supply of 12VDC. They can be controlled from the computer or programmed by the computer and then, disconnected to allow them work independently after being programmed (Monk, 2012). Meanwhile, Arduino was introduced in 2005 to provide an inexpensive and easy way for hobbyists, students and professionals to create devices that interact with their environment using sensors and actuators. Based on simple microcontroller boards, it is an open-source computing platform that is used for constructing and programming electronic devices which is capable of acting as a mini computer just like other microcontrollers by taking inputs and controlling the outputs for a variety of electronics devices (Leo, 2016). Meanwhile, there are different types of Arduino board, but the differences are based on the number of inputs and outputs (the number of sensors, LEDs, and buttons you can use on a single board), speed, operating voltage, form factor etc. Some boards are designed to be embedded and have no programming interface (hardware), which you would need to buy separately. Moreover, some can run directly from a 3.7 V battery, others require at least 5 V of power supply operate. However, these conventional types of development boards only allow users to access and manipulate microcontrollers for building embedded devices without knowing how exactly the microcontrollers relates to other functions which can be utilized, such as the BOOLEAN and TFT response and more than a single UART pins on same microcontrollers on the board in Plate 1. In addition, parts of its SPI pins are also identified as PWM pins which can be confusing for most students to understand for proper use and application.

Whereas, microcontrollers in Plate 2 are permanently soldered, which using them is not cost effective since the microcontrollers can get damage during practice, thus rendering the entire board useless.

Plate 1. Arduino UNO

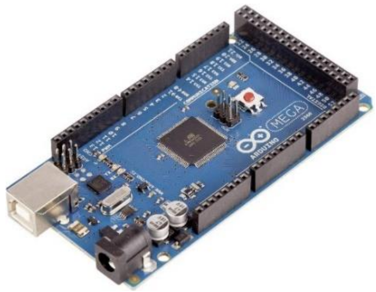

Plate 2. Arduino Mega2560

The NI myRIO is an embedded hardware device as shown in Plate 3, designed and developed specifically to help students design real, complex engineering systems, and as a controller, it is a portable reconfigurable I/O (RIO) device which students can use to design controls, robotics, and mechatronics systems. The myRIO Embedded Device features in/out pins either on one or both sides of this device in MXP and MSP connectors, which includes analog inputs, analog outputs, digital I/O pins, LEDs, push button, a Xilinx FPGA, onboard accelerometer, and a dual- core processor, while the user can program it with LabVIEW or C (Rob *et al.*, 2017). The MXP port has 16 Digital Input-Output, 2 Analog Output, 4 Analog Input and UART communication channels, and the channels can also be configured to accept an Encoder input, 3 PWM outputs and SPI and I2C communication. While, the MSP (the C port) has 8 Digital Input-Output, 2 Analog Output, and 2 Analog Input channels. Although, the C port can also accept 2 Encoders (Pokhsraryan, 2016). Furthermore, a shield was later developed as shown in Plate 4, which categories the MXP ports to an Arduino UNO pins layout. However, the microcontroller on the NI myRIO board as shown in Plate 3, are permanently fitted also, in which students cannot experiment external microcontrollers on the same board, thus, the boards are rendered useless to when the microcontrollers go bad. Likewise, the effect of changing clock either from a lower to higher value cannot be easily done without other components being damaged.

Plate 3. NI myRIO

Plate 4. Shield Adapter for NI myRIO

The Raspberry Pi (Plate 5), was officially created in February 2012, and the developing stone was laid by Raspberry Pi Foundation and within 3 years with the intention to promote the study of computer science and related topics in schools and in developing countries. It is powered by an ARM-based processor, which operates on 700 MHz–1.2 GHz, with a memory of 256 MB–1 GB, depending on different models (Zhong and Liang, 2016). The Raspberry Pi board contains Broadcom based ARM Processor, Graphics Chip, RAM, GPIO (consisting 2-Data Pins, 5-SPI Pins, 2-12C pins), Micro-USB Power Supply, SD Card Slot, USB Ports & Ethernet Port, HDMI (High-Definition Multimedia Interface), Video Out (RCA Cable), Status Led's, CSI Camera Connector and DSI Display Connector (Nayyar and Puri, 2015).

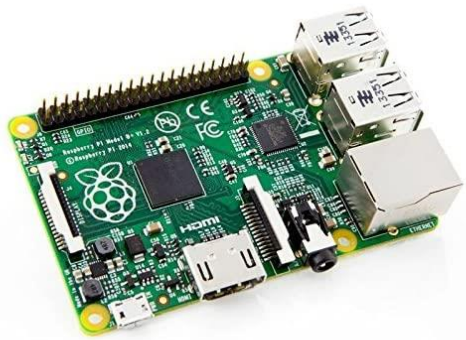

Plate 5. Raspberry Pi Board

The main idea behind the development was to give kids tiny and affordable computer in the period where computers were expensive and programming practice among kids was not supported by parents of children in U.K (Nayyar and Puri, 2015). However, although, they have more flexible connection ports but yet, this type of conventional development board was not also built for student to explore and understand the potentials of a microcontroller, rather to experience a mini computer which can also serve as an embedded device at the same time with an irreplaceable microcontroller, and also not cost effective. These existing platforms are needspecific platforms and do not meet all the requirements. Students and beginners find it difficult to apply their goal concepts using those boards because of the specialized nature of implementing the add-ons and basic peripherals, hence the crafting of the Novel Prototyping Development Board to offer some good value with respect to cost, expandability, quality enhancement of teaching and ease of use and also to find some local solutions to the challenges of Industries, Colleges and Universities.

MATERIALS AND DESIGN METHODS

The Printed circuit board (PCB) for the Novel Prototyping Board, is a double-sided circuit connection, whereby, the top layer circuit connections are colored RED, while the bottom circuit connections are coloured differently (Figure 1). Figure 2 depicts the printed circuit board with the bottom circuit connections colored GREEN. The circuit is designed using PROTEUS ISIS VSM, which can be converted into DXF file for drilling and artwork for circuiting printing. The Novel Prototyping Board was designed and implemented with a PCB software of an open-source prototyping development board, with microcontrollers like PIC18F886, dsPIC, ATMEGA328A and serves as the main controller or central processing unit. This board can connect to PC via USB and communicate with the IDE. It has extensions for analog signal input/output ports, digital signal (IIC, SPI, USART and BOOLEAN) input/output ports.

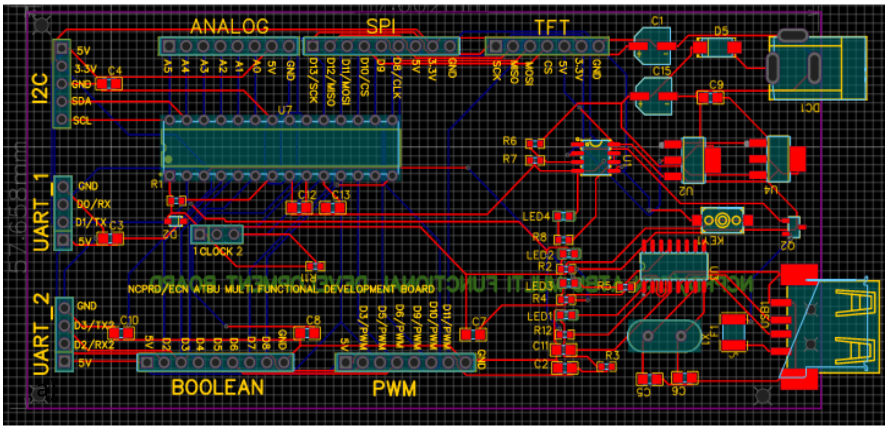

Figure 1: Printed Circuit Board for the novel Prototyping Development Board (Top layer circuit connections are colored RED)

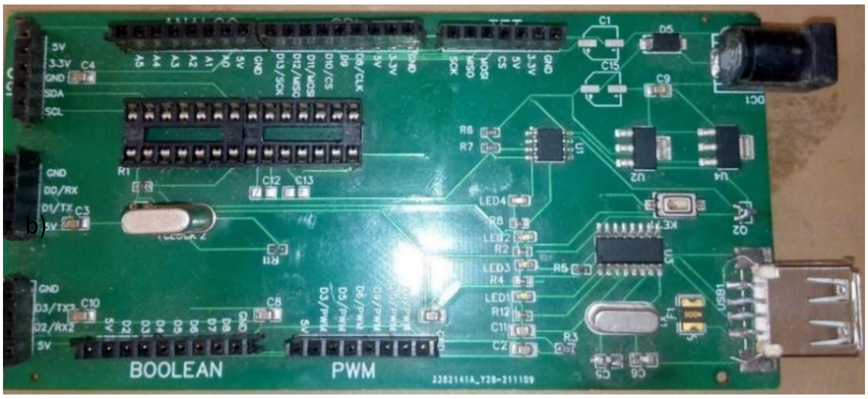

Figure 2: Printed Circuit Board for the novel Prototyping Development Board (Bottom circuit connections are colored GREEN)

The Schematic Circuit Design of the novel Development Board designed with ISIS Proteus which is an Intelligent Schematic Input System is shown in Figure 3. The Double Face PCB Schematic

Design for the novel Development Board Using ISIS is shown in Figure 4. It depicts the circuit connections between the components using PROTEUS ISIS VSM. All signals are categorized based on types using a Single In-Line (SIL) Female header pins to distinguish them between themselves and for ease of identification by the user. The schematic circuit design is then simulated for possible circuit printing. The circuit shown in Figure 4, is a simulated circuit drawing and components positioning of the schematic circuit design as shown in Figure 3.

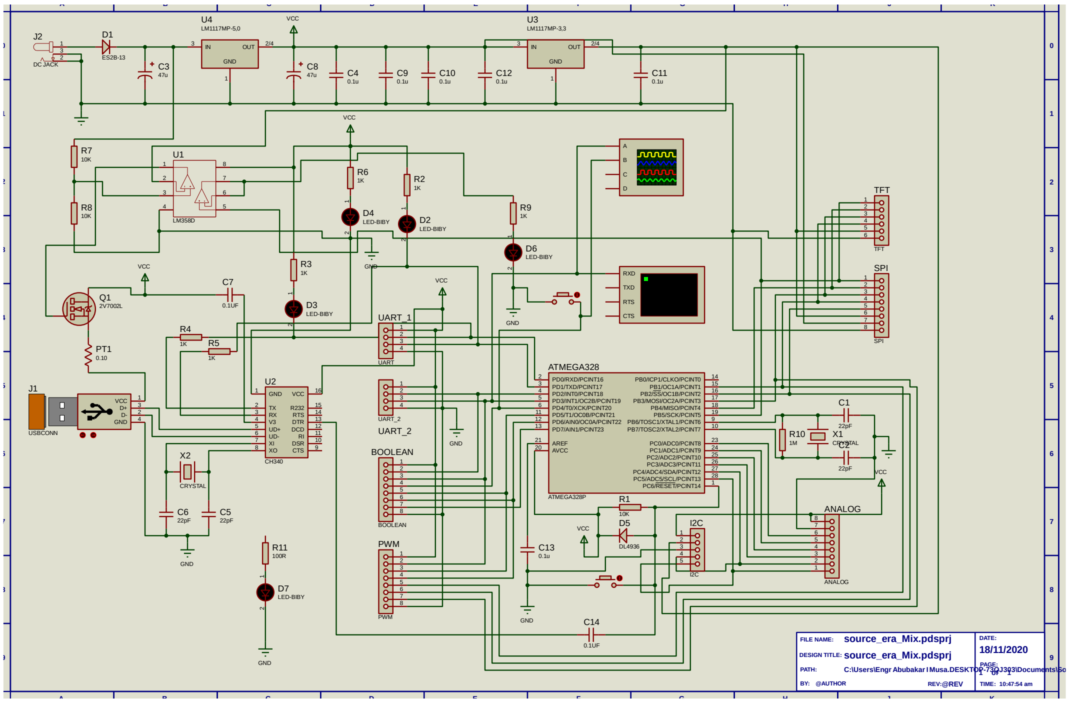

Figure 3: Schematic Circuit Design of the novel Prototyping Development Board using ISIS

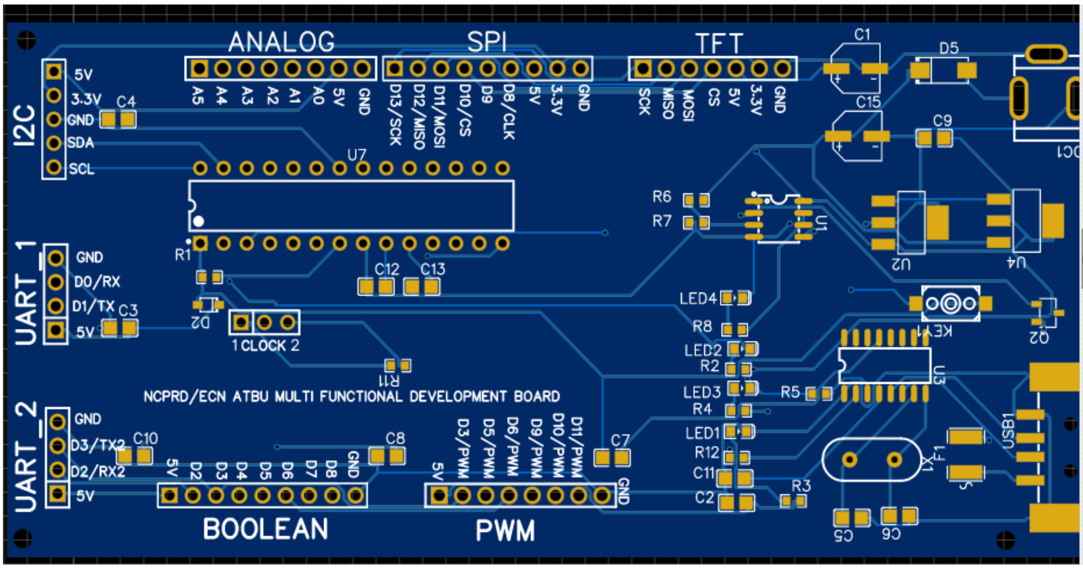

Figure 4: Double Face PCB Schematic Design for the novel Development Board using ISIS

RESULTS AND DISCUSSIONS

The Simulation Results of the novel Prototyping Development Board is shown in Figure 5. The Printed Circuit Board Model of the Novel Prototyping Development Board as shown in Figure 6 and consists of AM1117-3.3v (10), AM1117-5.0v (9), TFT-SIL Female (1), SPI-SIL Female (2), Analog-SIL Female (3), I2C-SIL Female (4), UART_1-SIL Female (5), UART_2-SIL Female (6), BOOLEAN-SIL Female (7), PWM-SIL (8), CH340 Chip (15), 16MHz Crystal Oscillator (12 & 13), Microcontroller (14), Data Transfer LED (17), Data Receive LED (16), Type-B USB Port (11) and 6-12V DC Jack (19).

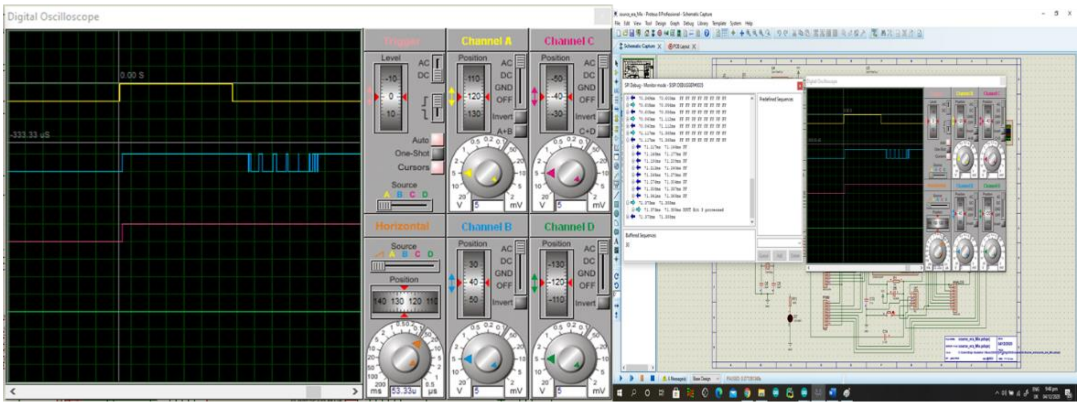

Figure 5: Simulation Results of the novel Prototyping Development Board

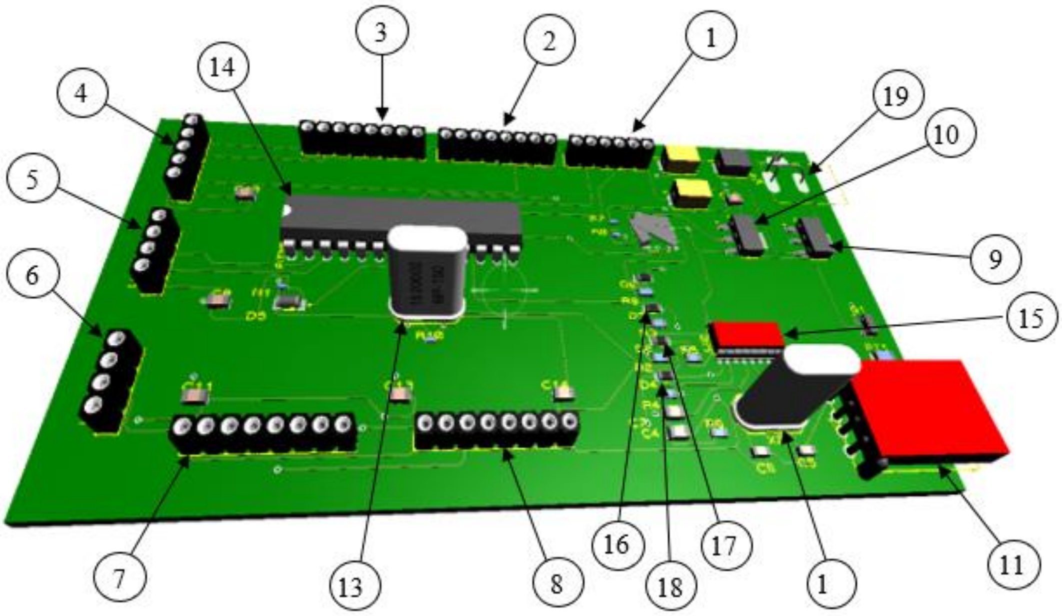

Figure 6. Printed Circuit Board Model for ERA Prototyping and Development Board using ISIS

The Source-Era Development Board operates majorly on 5V which is regulated by the AM1117-5.0V (9) from the 6-12V DC Jack (19). The regulated 5V is stationed to power the Microcontroller (14) and the VCC required on all the signals and Power LED (18) comes on which indicates the Source-Era Development Board is powered. Meanwhile, AM1117-3.3v (10) also regulates the 5V power supply to 3.3V for external use, such as microcontrollers, components and devices running on 3.3V logic. The Microcontroller (14) is not permanently connected on the Source-Era Development Board, but inserted in a 28 pins microcontroller socket so that other microcontrollers can be inserted for programming without having both the Source-Era Development Board and the microcontrollers being damaged. A Type-B USB Port (11) which is used for programming the Microcontrollers (14) also serves as a 5V power supply to the Source-Era Development Board. Although, a CH340 Chip (15) is connected to the Type-B USB Port (11) to interface the computer and the programmable Microcontroller (14). Similarly, the CH340 Chip

(15) is also powered by the regulated power supply from the AM1117-5.0V (9) and the 5V power supplied by the Type-B USB Port (11) when connected to the computer only. The 16MHz Crystal Oscillator (12) is permanently fixed on the board and connected to the CH340 Chip to provide stable clock due to its high frequency reference. While the 16MHz Crystal Oscillator (13) is also connected to the Microcontroller (14) for clock stability. However, the 16MHz Crystal Oscillator (13) is not permanently fitted on the board which gives room for inserting different Crystal Oscillators with different frequency levels, in order to evaluate their effect to all the signals generated by the Microcontroller (14). In addition, all the signals are categorized based on type and are utilized via a female header pin as the TFT-SIL Female (1), SPI-SIL Female (2), Analog-SIL Female (3), I2C-SIL Female (4), UART_1-SIL Female (5), UART_2-SIL Female (6), BOOLEAN-SIL Female (7) and PWM-SIL (8) which enables the students to easily know where their expected results can be.

CONCLUSION

A working multi-functional development board has been designed and successfully tested on various products. The device has the ability of interfacing a controller with different input and output device based on their signal type which include universal Asynchronous Receive and Transmit (UART) Protocol, Serial Peripheral Interfacing (SPI) Protocol, Inter Integrated Chip (IIC or I2C) protocol, Analog signal processing protocol (ASP), Boolean Digital, Pulse width Modulation (PWM). The device when properly packaged, can be helpful to researchers, student and industries in the field of electronic design, coding, and prototyping.

Symbols

UART: Universal Asynchronous Receive and Transmit Protocol,
SPI: Serial Peripheral Interfacing Protocol,
IIC or I2C: Inter Integrated Chip protocol,
ASP: Analog signal processing protocol
PWM: Pulse width Modulation
PCB: Printed circuit board
DSP: Digital Signal Processors
RTOS: Real time operating system
USB: Universal Serial Bus
PWM: Pulse-width modulation
SPI: Serial Peripheral Interface pin,
MXP: myRIO Expansion Port connectors.
MSP: Mini System Port
IDE: Integrated Development Environment
PC: Personal Computer
ISIS: Intelligent Schematic Input System

REFERENCES

[1]. A. Nayyar and V. Puri. Raspberry Pi-A Small, Powerful, Cost Effective and Efficient Form Factor Computer: A Review. Int. J. of Adv. Res. in Comp. Sci. and Software Eng. Volume 5, Issue 12, pp. 720-737. December 2015.

[2]. A. S. Patil and U. J. Tupe. Recent Trends in Platforms of Embedded Systems. Int. J. of Creative Res. Thoughts Vol. 8, Iss. 11; November 2020

[3]. D. Pokhsraryan. Fast Data Acquisition system based on NI-myRIO board with GPS time stamping capabilities for atmospheric electricity research. 5th Int. TEPA Symp. october, 2015; pp. 23-27.

[4]. G. Phipps. Comparing observed bug and productivity rates for Java and C++". Software-Practice and Experience. 29(4): 345–358; 1999.

[5]. H. Chen. "Comparative Study of C, C++, C# and Java Programming Languages". Vaasan Ammattikorkeakoulu University of Applied Sciences Degree Program of Information Technology, pp 4-78; 2010.

[6]. L. Leo. Working Principle of Arduino and using it as a Tool for Study and Research. Int. J. of Cont., Auto., Com. and Sys. Vol.1, No.2; pp. 21-29. April, 2016.

[7]. M S Abubakari and Mashoedah. The Internet of Things (IoT) as an Emerging Technological Solution for the Covid-19 Pandemic Mitigation: An Overview. J. Phys.: Conf. Ser. 1737 012003; pp. 1-9. 2020.

[8]. S. Hollenthoner. Micro-Processors and Micro-Controllers. A publication of IAEA Laboratories, Seibersdorf, Austria; pp. 6-7; January 2007.

[9]. S. Monk. Getting Started with Sketches in Programming Arduino - Getting Started with Sketches. The McGraw-Hill Companies, 2012.

[10]. X. Zhong and Y. Liang. Raspberry Pi: An Effective Vehicle in Teaching the Internet of Things inComputer Science and Engineering. Electronics. 5. 56.pp. 1-9. September 2016.