Electronics & Communication Engineering Department, IIITDM KANCHEEPURAM April -13, 2017,India

भारतीय सूचना प्रौद्योगिकी अभिकल्पना, एवं विनिर्माण संस्थान, कांचीपुरम
Indian Institute of Information Technology, Design and Manufacturing, Kancheepuram
(An Institute of National Importance Fully Funded by Govt of India)

Arduino Based Virtual Instrumentation Suite (Ar-VIS)

Aparnaa R, R Krishnakumaran, Kiran kumar R, Rajesh Laguri

edm14b003@iiitdm.ac.in, edm14b024@iiitdm.ac.in, edm14b033@iiitdm.ac.in, edm14b034@iiitdm.ac.in

***Abstract*—Sensors are an extensive part of any engineering program. Sensory data processing is rather versatile work, owing to the variety of sensors involved and variations in their structure, like the pin arrangement, for example. A sensor DAQ and processing kit, should not only accommodate these variations, but also make system power efficient and user friendly. The currently used sensor-module suite & instrumentation kit, NI-ELVIS, is highly expensive and proprietary, which limits the access to students. Also the sensors are hard-wired which makes sensor failure a costly affair. The main aim of AR-VIS is to counter these disadvantages. It aims to provide a less-expensive user friendly GUI, with a DAQ setup that focuses mainly on codeless programming.**

Keywords— Data acquisition, Processing, sensory network, amplifiers

1.INTRODUCTION

As a part of Engineering courses, Sensing and Control labs require students to learn and use various specialized sensor kits and compatible Analysis tools. Some of these kits are too expensive, and hence not easy to repair when needed. Moreover the software tools used for analysis, although undeniably versatile and robust, require time to learn, to be used with full efficiency. They involve a certain amount of syntactic and semantic comprehension, which could only be acquired over time. Further as the sensor kits pack on-board all the sensor modules with the required bias, impedance matching and amplifiers, there is too little transparency for the students to appreciate the complexity of the sensor interface. Hence this project aims to build a Sensor lab training kit, that is less expensive, easy to build and repair, and offers the user complete insight of the working of the kit. The product is modularised for fault isolation and repair purposes, and also so the sensors used have no hardwired constraints as in the conventional design. The product further aims to provide facility for wireless sensor network, as an endeavour to further modularise the product and improve its usability and fault correction capability.

2.Motivation/Outcomes

The ELVIS kit used in lab has always remained a black box, with its internal circuitry unexplored. This project enabled us to gain an insight into the circuit behind this sensor suite. The arrangement of sensors, on this board and also the data multiplexing for processing were also looked into, and helped us in framing the product in the most optimum way possible. Owing to this project, we also explored the language structuring involved in the GUI of the NI-LABVIEW, in order to try and recreate the efficiency it to the extent possible.This project also made us learn and get proficient enough in *Processing*, as the coding in the project for the GUI has been done in the said language. Our knowledge about various IC's and sensors have also improved, due to the extensive study done, for the purpose of building oscillators, amplifiers, function generators etc, The project also helped learn the usage of Autodesk Eagle PCB Design Software.

3.LITERATURE SURVEY

The students of the 2012 batch of Electronics branch, IIIT-D&M, Kancheepuram, had explored the possibility of creating a Function generator using IC 555.
The work proved to be relevant, due to the involvement of sensors,

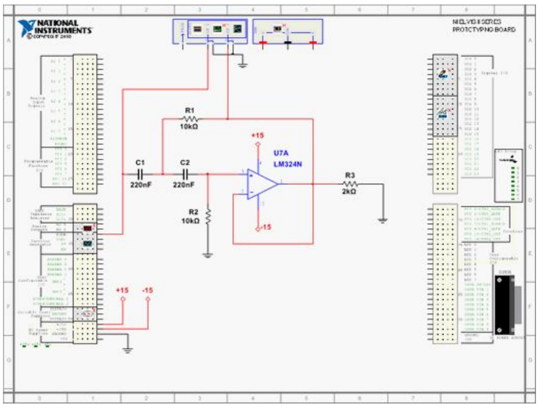

Fig 1 : NI ELVIS Proto Board Circuitry

The survey for our work consisted mainly of understanding the intricacies involved in the working of NI ELVIS Board. This was achieved by multiple readings of the relevant portions of the user manual of NI ELVIS. The other relevant studies included the data sheets of various sensors, to gain understanding of their physical structure and working. Other readings included the various processing and eagle PCB tutorials that enabled us to use the softwares conveniently.

4. TOOLS USED

A. Arduino

It is an open source, microcontroller hardware and software which is very commonly used by students for making projects. Online forums makes it easier for beginners to learn. Arduino boards are available commercially in preassembled form, or as do-it-yourself kit. This made it , a product of our concern.

B. Eagle PCB

EAGLE is a scriptable electronic design automation application with schematic capture, printed circuit board layout, auto-router and computer-aided manufacturing features. EAGLE stands for Easily Applicable Graphical Layout Editor.

EAGLE contains a schematic editor and a user-friendly PCB layout editor. EAGLE supports auto-routing facility.

EAGLE saves Gerber and PostScript layout files. These standard files are accepted by many PCB fabrication companies which made us choose this software for making the LAYOUT.

C. Processing

It is an open source computer programming language and integrated development environment (IDE) built for the electronic arts, new media art, and visual design communities with the purpose of teaching the fundamentals of computer programming in a visual context, and to serve as the foundation for electronic sketchbooks.
It was used to create a GUI where we can choose different options which will automatically write an arduino program (in .ino file format) directly in the sketchbook folder of the processing software.

5. METHODOLOGY

A. DAQ *Shield:*

1) Sensor *Port Layer:*

Further the positive power supply line can be switched between 3.3 V and 5 V. The external power supply pin of the Arduino has been connected to a pin on this layer. The Arduino could be powered by a power source such as a battery, up to 9 V. This, along with the Bluetooth port, allow Wireless Data Acquisition.

The top most layer of the Arduino DAQ shield (Figure 1) contains: ports for Analog and digital sensors or modules, a HC04 Bluetooth module port. The arrangement of the input, Vcc and ground pins is made such that commonly-used sensor modules for Arduino can be plugged directly onto the shield. To allow the use of other non-compatible sensor modules, a breadboard arrangement would be provided on this layer.

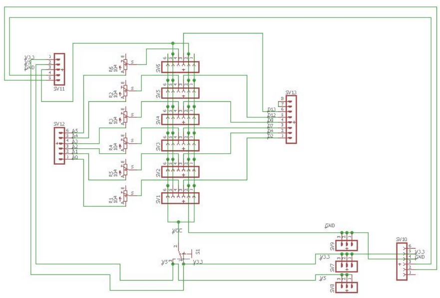

Fig 2 : Schematic of the top layer on Eagle PCB

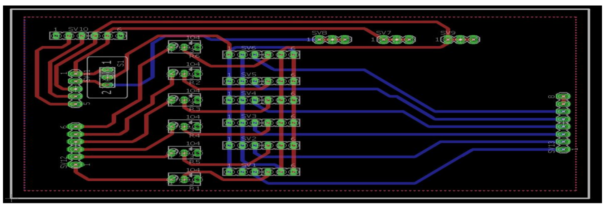

Fig 3 : A screenshot of the layout of the top layer

2) *Amplifier Layer:*

This is the second layer of the DAQ shield. Most analog sensors require a DC bias for their operation, and they output signals in the order of milliVolts, in addition to the DC offset. The ADC of the Atmega-328p IC used by the Arduino uno is 10 bits and has a range of 0 to 5 V. Hence the minimum voltage readable in the Arduino ADC is 5 mV.

The sensor output has to be amplified and brought to the range 0 - 5 V. The schematic of the amplifier is shown in Figure 4. The schematic implements the amplifier for a single analog sensor. The Amplifier is an Op-Amp based Difference Amplifier with Gain controllable by varying the PWM applied to the nMOS switches The PWM is applied from the Arduino PWM pins. This allows the gain of the amplifier to be controlled by software, even allowing for an automatic gain adjustment during runtime.

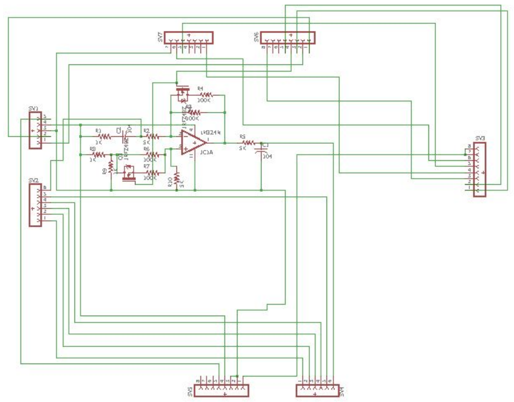

Fig 4 : Schematic of the top layer on Eagle PCB

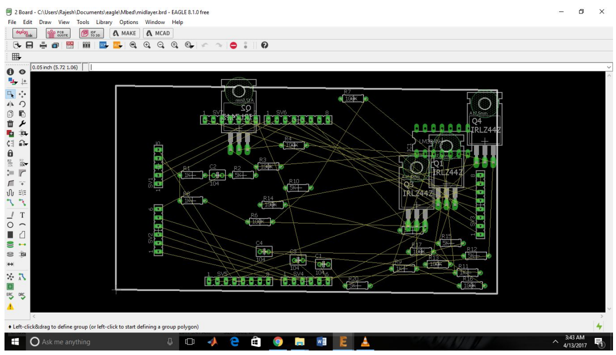

Fig 5 : A screenshot of the layout of the top layer

B. Logic Analyzer and Oscilloscope:

Written in Processing, the oscilloscope allows to select transmitted analog value to be plotted in real-time. For easy user experience, the timescale of the oscilloscope can be scaled and unscaled by pressing any key on the keyboard. The outputs of the digital pins are plotted according to channel selected by the user. There are six channels in total which can plot the digital pins 2 to 8.

C. Arduino Codeless Programming:

(Figure 8). The project name would be the Arduino code file name. The project description would be included in the Arduino code file at the top in block comments. Selecting Digital inputs, selecting Analog inputs, Enabling Serial with specified baud rate. Elaborate the above in 2 lines. E.g. Selecting serial baud rate of 9600 adds code **"Serial.begin(9600);"** in the setup() function. Digital input n(pin number) adds **"pinMode(n,INPUT);"** in the setup() and **"DigitalRead(n);"** in the loop() function. The digital inputs are transmitted by multiplexing onto a single byte and analog input readings are scaled down to range 0 to 255 and are transmitted as individual bytes in the pin numbering.

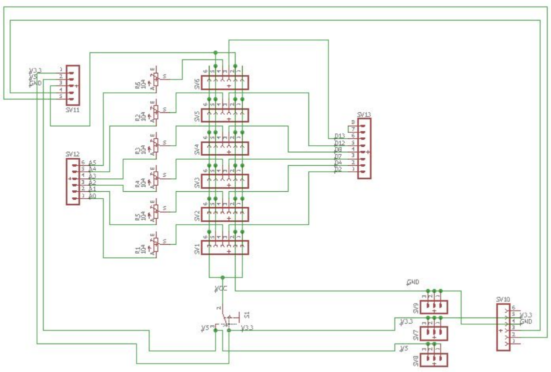

Fig 6: Schematic of the top layer on Eagle PCB

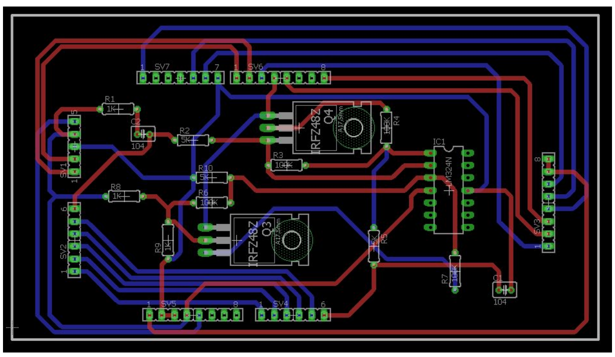

Fig 7 : A screenshot of the layout of the top layer

Fig 8 : A screenshot of the GUI

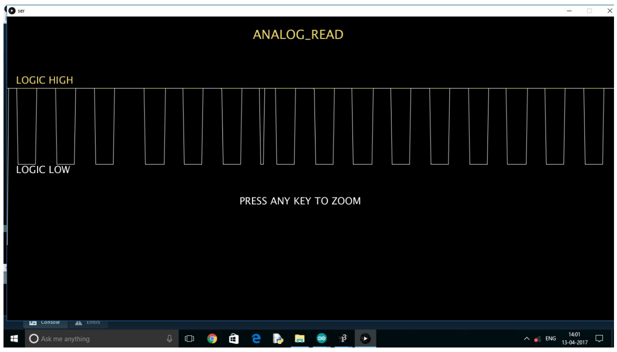

Fig 9 : A screenshot of the oscilloscope at work

6. SCOPE OF IMPROVEMENT

The Product functionalities can be further improved to accommodate more sensors. The wireless sensor network can be further perfected by introducing provision for access using Wi-Fi. Rotatory pins can be created so that the pin arrangement can be changed as per the sensors used. The GUI can further be made user friendly. Additional facilities like Bode Plot analyser, etc, can be made available.Oscilloscope's plotting speed could be improved. Further the six channel oscilloscope can be extended in the future design to include all analog and digital pins.

7. RESULTS

The GUI was tested as an interface to arduino UNO. The oscilloscope responded appropriately as per the pins connected. It displayed logic high voltage when connected to the 5V of the arduino, logic low voltage when connected to ground and PWM pulses when connected to a when PWM was activated in arduino pin

11 (Fig.9).

8. CONCLUSION

The project aim was to build a sensor and instrumentation kit that enabled a modularised working of the sensory output processing. Codeless programming was attempted in order to make the interface more user-friendly for the beginners. The intended objectives of the product were achieved and the product working was tested in an appropriate environment.

9. REFERENCES

[1]Processing Tutorials:

https://processing.org/tutorials/overview/

[2]Eagle PCB:

https://learn.sparkfun.com/tutorials/using-eagle-board-layout

[3]Processing Tutorial:

www.hrenatoh.net/curso/processing/processing_benfry.pdf

[4]Eagle PCB:

.

https://intranet.ee.ic.ac.uk/t.clarke/EAGLE/The%20EAGLE%20Guide.pdf