The Sweet Sound of High-Quality Microphone Testing (Validation Teststand for Microphone Testing)

by
Brad Westfall
Senior Engineer
Data Science Automation, Inc.
USA

Category:
Design R&D
Electrical/Electronics

Products Used:
NI LabVIEW 8.2.1
NI Sound and Vibration Toolkit 4.0.49153

The Challenge:
The customer needed a software application to validate a microphone's acoustic performance. The application had to be able to communicate, acquire and process data from as two separate analog or digital microphones.

The Solution:
Data Science Automation met this challenge by leveraging a modular LabVIEW-based software architecture with USB acquisition devices. The data processing was accomplished using LabVIEW's Sound and Vibration toolkit.

Abstract:
A microphone teststand manufacturer needed software to validate proprietary microphone acquisition hardware. In addition, the application required a unique hardware configuration to match the microphone teststand's unique physical structure. The system was implemented so that it was flexible enough to support present and future teststand designs.

Overview
Few people realize how often microphones are being integrated into every day products. State of the art digital microphones incorporate solid state circuitry that significantly reduces their size and cost (Figure 1).

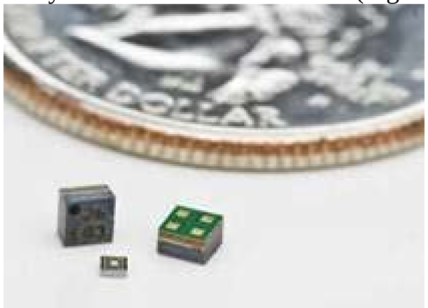


Figure 1. Digital Microphone.

Although at a high level all microphones are the same, most new applications come with design challenges. For example, most laptops have an integrated microphone. Sometimes the background sounds overwhelm the desired sounds. Similarly, VOIP (Voice Over IP) applications allow voice communication via an internet connection but the advantages of such applications are less valuable if sound quality is poor. To begin addressing these sorts of problems, a microphone teststand manufacturer designed a new microphone configuration and custom USB acquisition board that could validate the performance of two microphones simultaneously. The teststand supported both digital and analog microphones.

However, in addition to the inherent complexities of the acquisition and data analysis processes, the user interface was complex as well. For example, it had to present data in a format that would be easily understood by engineers with a variety of backgrounds. To meet all these requirements the customer called-in Data Science Automation. The result was a software application that with only a few mouse clicks could simultaneously acquire, analyze and clearly present the validation of multiple microphones.

Approach

The application's user interface had to be aesthetically pleasing while maximizing functionality. Likewise, the software design had to provide flexibility while constraining scope to ensure reliable results for all engineers within the budgetary limits of the effort. To accomplish this, Data Science Automation architected and developed a modular application with high cohesion within modules and low coupling between modules (Figure 1).

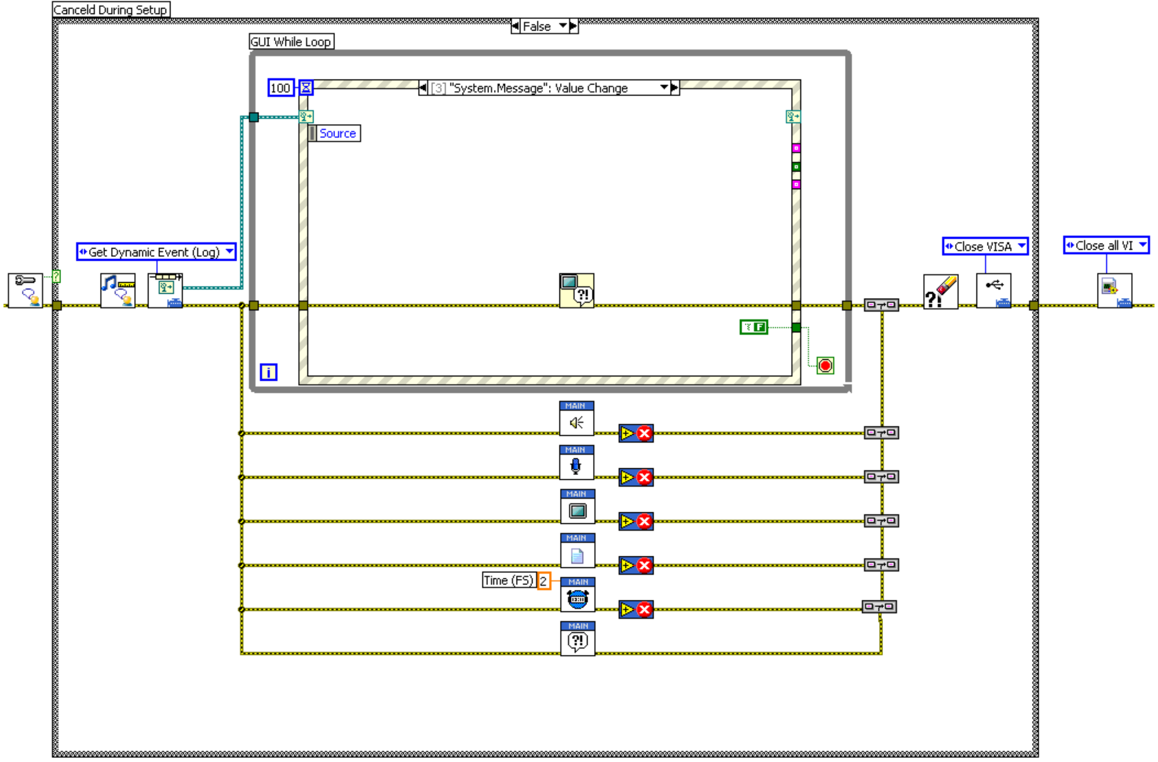


Figure 2 - Block Diagram of main application demonstrating the modular architecture

To complete a microphone validation the software application had to:

- respond to user commands,
- generate sound,
- acquire sound,
- display results,
- log data, and
- monitor time.

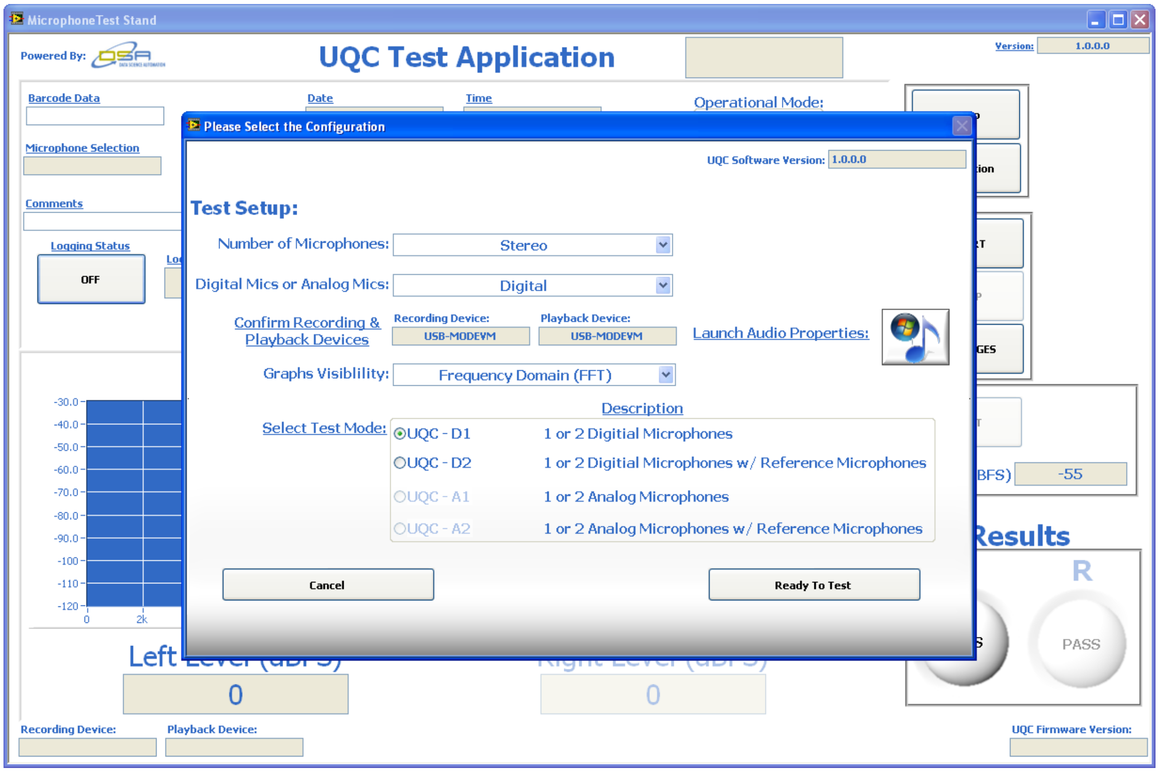
To achieve each of these tasks Data Science Automation employed a well designed and thought out architecture that ensured each module or task had all the information it needed to operate while limiting the degree to which modules had to rely on data from other modules or tasks.

Sound Analysis

The main purpose of the application was to test, analyze and validate microphones. To achieve this requirement Data Science Automation incorporated the Sound and Vibration toolkit from National Instruments. This toolkit facilitated the ability to acquire large datasets from a microphone and analyze the values that indicate failure of the validation test. Now to process the acquired microphone sound data, the application also had to produce a known sound with specified frequency and amplitude. The application met this requirement by taking advantage of LabVIEW's ability to easily and efficiently communicate with USB devices – in this case, attached speakers. The ability to both produce and analyze sound from within the same application development environment allowed Data Science Automation to save time, effort and cost while producing a highly efficient and reliable sound analyzer.

Configuration and Data Logging

To configure the teststand the user interface provides controls (Figure 2) for entering such setup details as the number of microphones under test, and their corresponding analog or digital format. The application also needed a mechanism for entering the advanced configuration parameters that would be needed during troubleshooting. These advanced configurations had to be secured to prevent unauthorized user access. This requirement was met by establishing password protection on restricted areas (Figure 3) and assigning user privileges based on passwords.



Version: 1.0.0.0

UQC Software Version: 1.0.0.0

Number of Microphones: Stereo

Digital Mics or Analog Mics: Digital

Recording Device: USB-MODEVM

Playback Device: USB-MODEVM

Graphs Visiblility: Frequency Domain (FFT)

Select Test Mode:

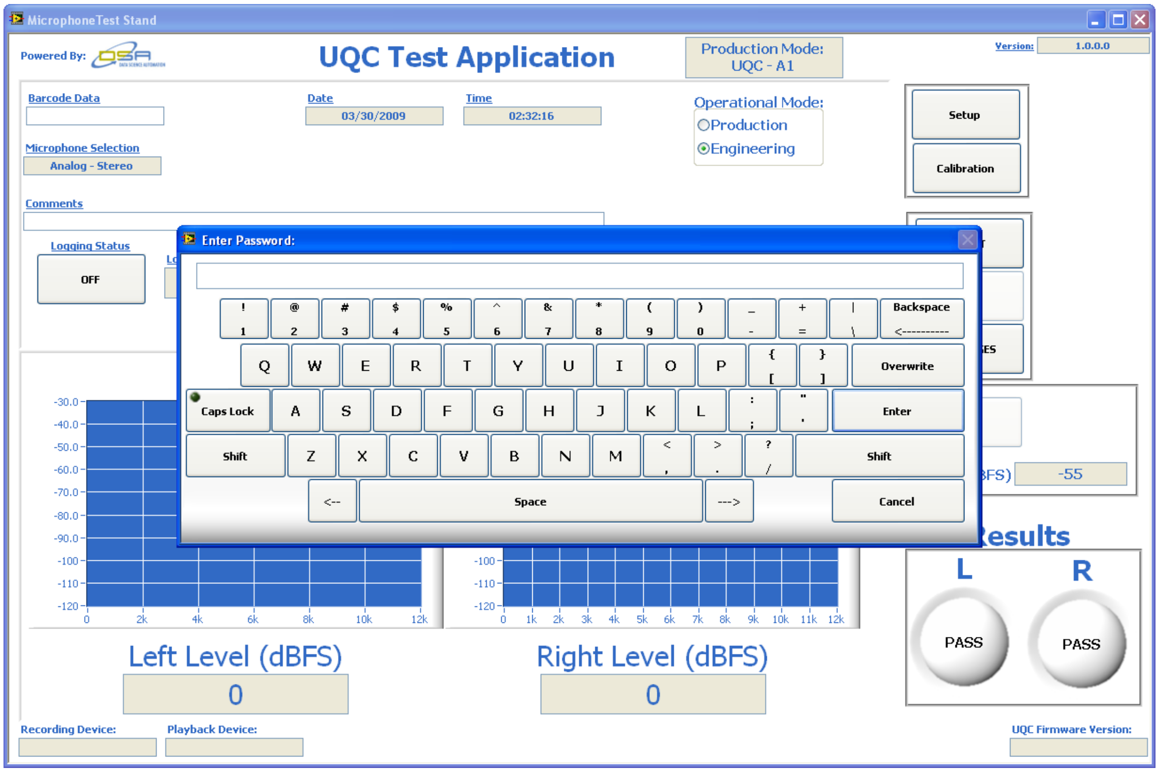
	Description
● UQC - D1	1 or 2 Digitial Microphones
○ UQC - D2	1 or 2 Digitial Microphones w/ Reference Microphones
○ UQC - A1	1 or 2 Analog Microphones
○ UQC - A2	1 or 2 Analog Microphones w/ Reference Microphones

Left Level (dBFS): 0

Right Level (dBFS): 0

(dBFS) -55

Figure 3 - User Interface using simple mouse clicks for configuring test parameters


MicrophoneTest Stand

Powered By:

UQC Test Application

Production Mode: UQC - A1

Version: 1.0.0.0

Barcode Data

Date 03/30/2009

Time 02:32:16

Operational Mode:
○Production
◉Engineering

Microphone Selection: Analog - Stereo

Comments

Logging Status: OFF

Enter Password:

Left Level (dBFS): 0

Right Level (dBFS): 0

(dBFS) -55

Results
L: PASS
R: PASS

Recording Device:

Playback Device:

UQC Firmware Version:

Figure 4 - User Interface for getting password to enforce user privileges (Using an on-screen keyboard supported flexible input media such as touch screens)

In terms of data collection, the system automatically collects all data for each user-defined test while storing the data into a file containing all processed data. A total of 27 parameters are recorded.

Display

The main purpose of the application's graphical display is to allow all operators ease of access to the information needed to determine if a microphone is functioning correctly. To simplify the code behind the graphical user interface, Data Science Automation incorporated a software architecture for displaying data that utilized LabVIEW sub-panels. These sub-panels allowed small areas to contain a multitude of statistics and graphs based on the user preferences and privileges while including a simple pass/fail indicator. The benefit of this structure to the customer is lower costs for maintaining and expanding the code over the application's lifetime.

The Bottom Line

The software system described in the paper met or exceeded the customers needs for a flexible and reliable teststand software package in large part due to the skill and imagination that developers at Data Science Automation were able to apply as a result of their in-depth knowledge of LabVIEW's native features.