Functional Test System Development Plan

# Introduction & Justification

This project will create an open-source, reusable software platform for bed-of-nails (pogo-pin) test fixtures. This will allow Sensit to (1) quickly develop bed-of-nails test fixtures without requiring software development, (2) increase productivity by automating manual labor for testing (3) enable accurate Pareto analysis datalogging of first-pass yield. This will ultimately decrease time-to-market for new products and enable accurate Pareto analysis for production.

Requirements

The deliverable is a Windows Forms App with the following abilities:

* Present the following user interface:
  + [insert screenshot]
* “Start” button: When the user presses the “Start” button, begin a series of tests in a new thread.
* “Stop” button: When the user presses the “Stop” button, if tests are running, abort the tests.
* Status bar: Display the status of test currently running.
* Progress bar: Display the percent progress of the series of tests. If no tests are running, display “Ready…”
* Menu bar:
  + File 🡪 Exit: Close the app.
  + Help 🡪 About: Display a form with the application version and repository URL.
* Tests should be defined by a configuration file, which represents a serializable class (see [insert file].
  + Configuration file defines a series of tests. Each test contains:
    - a command string (which will be sent via a serial port to a test device),
    - An expected response string,
    - A picture to be displayed to the user upon test failure,
    - A string displayed to the user upon test failure.
* All test results shall be recorded in a database.
* User must be able to set database location.
* User must be able to set serial port and device type (pick from devices in the SDK).

After the initial application is delivered, support for additional hardware such as multimeter, temperature chamber, etc. can be added to the SDK as needed, and will be automatically supported in the application.

# Development Plan

The application will be developed in C# using Visual Studio. Sensit is familiar with these technologies and can easily support them. If needed, a future project could upgrade these applications to run on a variety of platforms, including Android, Linux, or as web applications.

The application will be organized into three projects:

* A Software Development Kit (SDK) which contains utilities for each of the deliverables and can be reused in future applications.
* A WinForms application which utilizes the SDK to present the deliverables to the user.
* A collection of unit-tests for the SDK to aid development and maintenance.

To facilitate code reuse, the SDK, its unit-tests, and the application will be stored in one or more version-controlled repositories.

The deliverables do not require intellectual protection and will be released under an open-source MIT license. All proprietary information will be contained in application configuration files created by company technicians.

This test could be accomplished using the following Arduino-compatible relay board and DC/DC converter:

* [KTA-223 USB/RS485 Relay IO Board](https://www.sparkfun.com/products/9526) ($144.95, via SparkFun)
* [SparkFun Buck-Boost Converter](https://www.sparkfun.com/products/15208) ($9.95)

The KTA-223 has eight relay outputs, three analog inputs, and 4 digital inputs, as well as a small 5V power supply, so it can control LEDs, control power to a device, and measure voltages. It can also be programmed using the Arduino IDE, or controlled remotely via serial port. The Buck-Boost Converter can be configured to output any voltage between 2.5 – 9V, so can be used to simulate a lithium battery cell’s 3.6V output.

# Revision History

|  |  |  |
| --- | --- | --- |
| **Date** | **By** | **Change Description** |
| 4/5/2022 | Adam Johnson | Initial revision. |