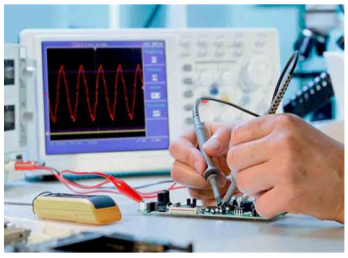


Designing a Semi-Automatic Final Functional Tester

By Carl Angotti, Angotti Product Development, https://www.angotti.com

Introduction

Semi-Automatic Final Functional Testers sit in the tester spectrum between a 100%-manual tester and a 100% automatic tester. Differences among them are described in the next section. The primary tradeoff is how much or little is available to spend on the tester, balanced against the impact of the device selling price and sales volume. The more automated a tester is, the lower the cost to perform the testing. Of course, a highly automated tester is the most expensive to create.

What is a Semi-Automatic Tester?

The various types of testers can generally be described as follows:

- **A manual tester** is dominated by the testing having the Test Operator almost exclusively doing all of the UUT (Unit Under Test) testing manually. The Operator follows a detailed written test procedure and manually records test results.

  Frequently, this tester style has the Operator set up the test equipment for each test and then read and record the results or indicates pass or fail for a particular test. There is also special cabling for such a tester, and sometimes a manual switch array is used to help speed testing. See the APD (Angotti Product Development) paper at https://www.angotti.com/docs/040921_EZine.html for an example of such a tester.

An automatic tester requires a minimum of Operator skill and involvement in the UUT testing process. This type of testing is characterized as having a minimum of Operator skill and involvement in the UUT testing process. It also produces the fastest and most controlled test environment. In addition, it provides the highest quality and repeatability of the testing. The reading and recording of test results are also minimized, thus reducing error rates.

The Operator test costs are minimal, and the test speed is very high. There is likely to be at least some form of manual testing involved in practice, even if it is only to read and acknowledge a display or LED output.

A Semi-Automatic tester sits somewhere between a fully manual and a fully automatic tester around the midpoint. There is a considerable manual

intervention of the Operator in the testing, including reading and verifying the test results. Often the results are guided via a PC program that records an archival file of the test results.

If the UUT contains one or several embedded controllers, the tester most certainly is a semi-automated one. For such a tester, Also, the Operator needs at least some skill to read and interpret the measurements.

The operator test costs for such testers sits between those of a manual and fully automatic tester. The test speed is slower, and the test quality is somewhat lower than an automated tester, but tester development costs are much less than with a fully automated tester.

Reasons for selecting a Semi-Auto approach

Such a test approach is warranted if the UUT has multiple microcontrollers plus lots of inputs and/or outputs. The PC test program can present commands to the system, and the Operator doesn't have to be burdened with manually entering these commands into the system. The Operator can then read the respective outputs to determine system test staus. Some of the outputs might be automatically monitored and tested for Pass/Fail criteria via the test program.

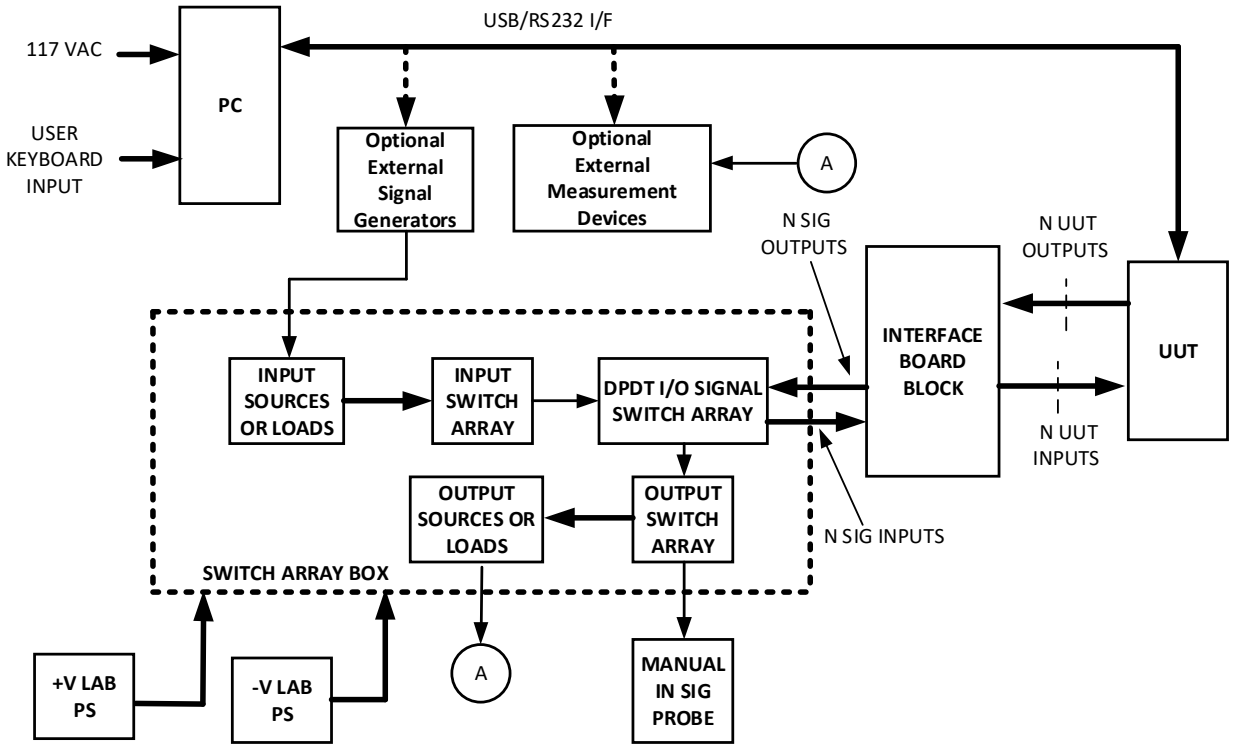
Sometimes, there is a need for considerable Operator intervention for a UUT that is dictated by the fact that insufficient critical test points have been brought to system-level connectors so they can be monitored easily. This lack of appropriate test points is usually brought about by insufficient DFT (Design for Testability) during the hardware design phase of the UUT.

Lack of Test Points necessitates manual probing by the Operator internal to the system. Further complicating testing is that sometimes there is no automatic way to assess the results of a test due to a lack of automatic instrumentation or the lack of ability for a test instrument to interpret the test results correctly.

Under the conditions mentioned above, a Semi-Automated tester is a correct choice.

Generic Semi-Auto Tester Block Diagram

The block diagram below illustrates a generic form of a semi-automatic tester. It shows an advancement over a manual tester. In this case, the PC communicates with an embedded controller in the UUT and can receive responses. The optional External Signal Generators or Measurement Devices can either be manually set and read or automatically set and read via a USB port. There is assumed to be some automatic and



OVERALL GENERAL Semi - Auto TESTER
BLOCK DIAGRAM
CARL ANGOTTI – 4/19/2021

some manual testing done by the test procedure running on the PC and some testing done by the test operator.

Semi-Auto Tester Block Diagram Example

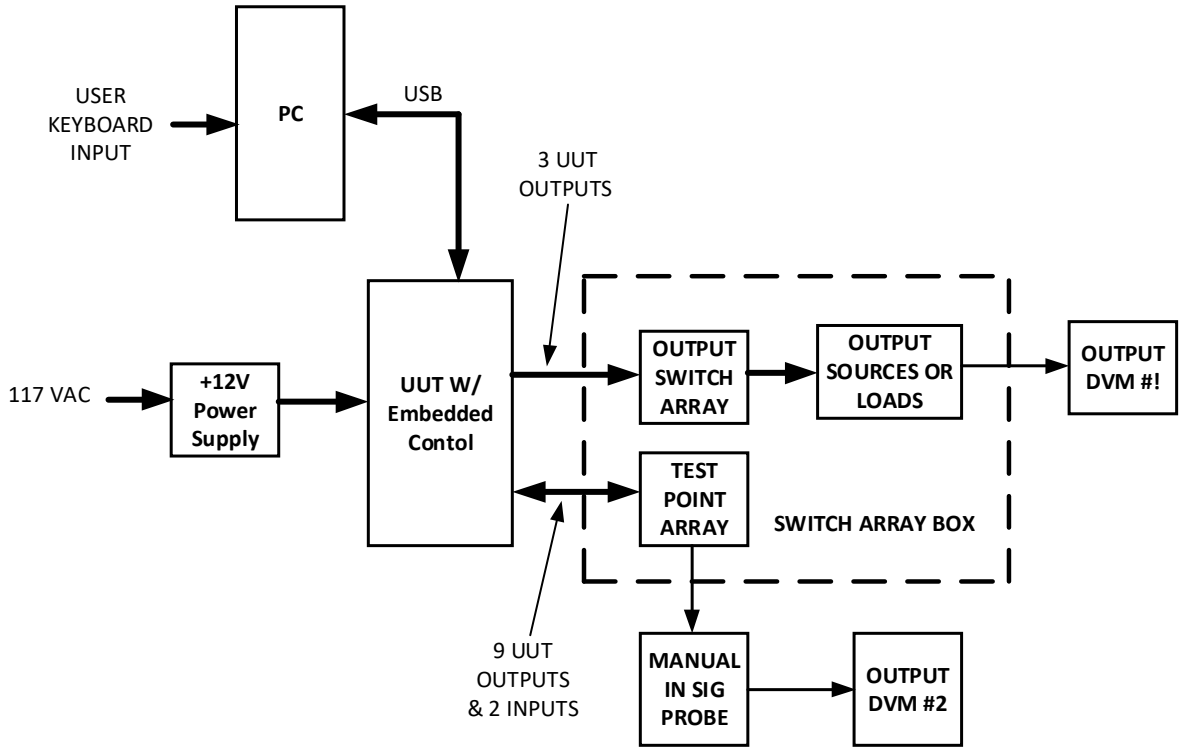
The block diagram below illustrates the system interconnections at the block diagram level of an example tester APD designed to test an in-company product. Note that it contains a simple switch array inside the switch array box. The system had only a few fixed signal inputs and nine measurement outputs. Some of these outputs are generated by the UUT, and some are done via hand probing.

The interface connectors and inputs come from a system interface connector custom-made for this UUT. All output signal measurements are done via one of two DVMs and are read by the Operator and recorded via the PC operator interface. This recording is accomplished in the same way described in the earlier Manual Tester article. The changes that occur are in the script code sent to the UUT.

This example tester is designed to meet the needs of a specific client. It was designed to test boards that have a maximum production rate of 15 per month.

Operator readouts were done via two auto-ranging DVMs. One monitored a voltage output, and another utilized a manual probe to verify internal voltages on the board and the switch array box.

A standardized test procedure facilitated the testing via a unique piece of software that guided the testing. This software allows the Operator to indicate pass or fail. A unique input text file was generated to guide test steps. This input text file utilizes a small set of script commands to serve as the procedure. The test results are then recorded in an archival output file for later review.



OVERALL SEMI- AUTO TESTER BLOCK DIAGRAM
CARL ANGOTTI – 4/18/2021

Illustration of the Test Interface Box

An illustration of the test interface box that was used appears below. It contains the input switch, along with the large test points used to test the UUT.

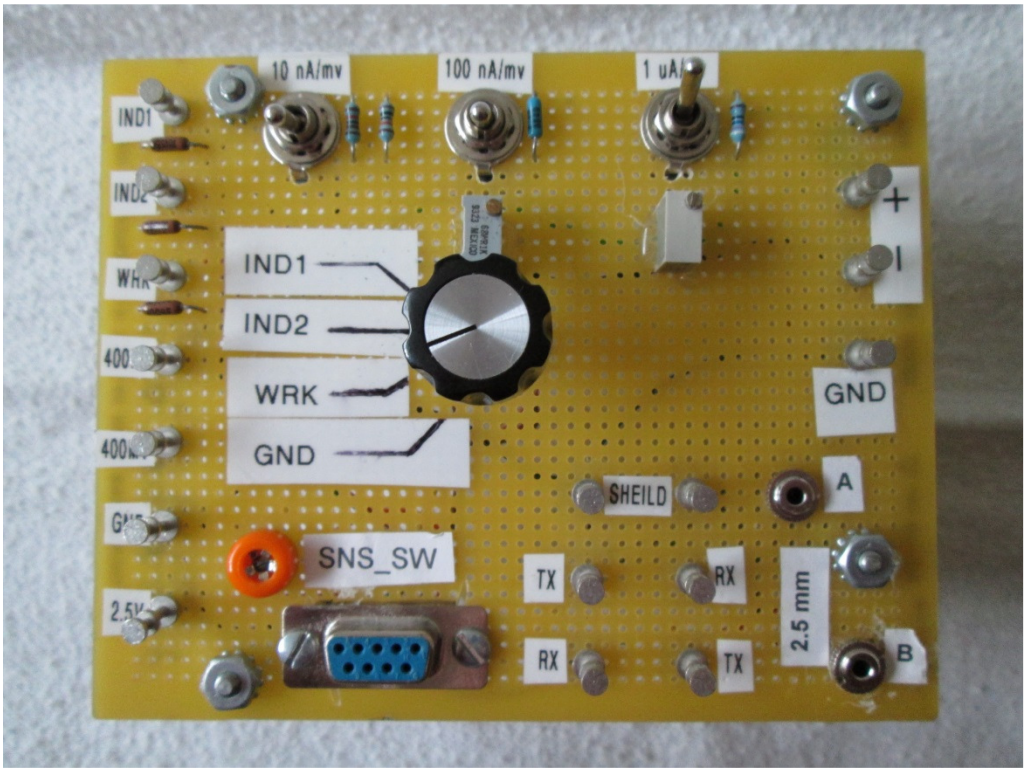


Illustration of the UUT under Test

A picture of the actual UUT System assembly under test using the actual tester is shown below. It shows the UUT connected via the RS232 style connector. There are three other UUT test cables in the test box. The tester also required an external test box used to supply a special voltage to the tester.

To the right are the two DVMs used to make the test measurements and read by the Operator.

To the left is the PC that communicates with the UUT via a USB to RS232 converter cable.

All command signals are sent over this USB port, and the Operator measures the results either by the PC display or on the DVMs. The Operator reads these, and answers pass or fail on the PC keyboard.

Semi-Auto Testing the UUT

The unit under test sequence using a Universal Basic test code executable is described here. Testing is accomplished utilizing a unique PC test control program. The test process is described in this section.

When the program starts, the program opens a window on the PCB. The program requests the name of the Operator and the UUT serial number on the PC screen.

After this, UUT testing proceeds by opening an input text file. Test instructions are presented to the Operator, and the results appear on the PC monitor. They are presented with the pass/fail criteria. The Operator then checks the results on the two DVMs and responds to the test pass/fail status or the PC screen. This response is then recorded on an output test archive file. After this, the subsequent instructions are presented, and the process continues until all of the tests are completed.

A snippet of the output file follows a snippet of the input test code for the UUT.

Input Test Code Snippet

The snippet below illustrates the form of the Input file commands from the text file that controls the program via the PC executable. It illustrates the simplicity of entering tests to the Operator for semi-automatic testing. This text file is the formalized Test Procedure that the Test Engineer creates. The code is broken into Sections (Groups) that can be skipped by the Operator if desired to advance more quickly to a particular set of tests.

To make the input text more understandable, comments were added to the actual procedure file. These comments appear in the snipped example below just to the right of the "<<. " Note: A complete set of test script commands is available by emailing carl@angotti.com.

```
'4/16/21 InputFile2                    << Comment in Procedure
```

```
' Final System Test
' Example Procedure Semi-Auto Test
' Open Comm Port For Testing
COM
' Functional Testing
' Using the Semi-Automatic
' Test Program, Perform the
' Testing up thru Calibration
' Test
'*******************************
# Section 1 Basic Tests              << Test Section #1
'*******************************
' Buzz out TX, RX and GND       << Test Described in Companion Test Procedure
' 2.5 mm Cable Test
? Do these tests pass?               << Pass of Fail Response from Operator
'****************************
' Delay 1 mS Test
'****************************
>DLY 1                               << Command to UUT Microcontroller
'****************************
' Read Empty File Profile
'****************************
>READ 1                              << Command to UUT Microcontroller
'****************************
' Read Test File Profile
'****************************
>READ 13
'****************************
' Read Help File Profile
'****************************
>READ 16
'****************************
' Perform Relay 1 Test
' SW Relay Closed
>RLY1 1
'****************************
' Perform Relay 1 Test
' SW Relay Open
>RLY1 0
'*******************************
# Section 2 Output Leakage Tests
'*******************************

CODE DELETED                                  << Actual procedure code removed

'*******************************
# Section 5 Wrk Calibration
'*******************************
' Set WRK Gain to FreeS Scale
' Observe LED I3 = ON
'****************************
>RLY2 1
'
' MORE CODE DELETED                           << Actual procedure code removed
'
'*******************************
# Section 6 Square Wave Tests Calibration
'*******************************
' Perform Square Wave Test
```

```
' Measure Voltage at TP1
' Enter Command Mode
' Enter SQWV 1 - Start Square Wave
' Hit Enter to Stop Square Wave
>SQWV
C To Continue, press Enter
' After Last Example Unit is Tested
' Quit this Program
' ***************************
# End of Test This UUT              << Last Section Marker
' ***************************
/END TEST THIS UUT                  << Command to End Test of this UUT
```

Output Test Code Snippet

The snippet below illustrates the form of the Output (Archival) file generated by the PC executable from the input file as testing proceeds. It illustrates what such a file might look like in practice during Semi-Auto testing. The output below is the file archive of the above Input File above run on the actual UUT.

Again, to make the input text more understandable, comments were added to the actual procedure file. These comments appear in the snipped example below just to the right of the "<<. "

```
'4/16/21 Test Menu File       >> File Comment Line
'Start Testing Menu - InputFile0
' ***************************
' Input Selection Menu
' 0 = Opening Menu - This One
' 1 = Initial Board Bring Up - No COM Port Open
' 2 = Align and Calibrate Test - COM Port ON
' 3 = Verify Test - Com Port ON
' 4 = Quick Verify Test, Skip Pot Adjust - Com Port ON
'***************************
' Select Input Number Desired
'***************************
' After Last Board is Tested
' Quit this Program
' ***************************
' Select a New File you want to open
% Get File # and Open         >> First Command Open File Selected
  Test Person's Name is: CAA  >> Name of Test Person printed from input
*****************************************   >> Printed by Test Program
*             Beginning of Testing          *
*        04/22/2021          13:38:18       *
*             UUT S/N = 5                   *
*****************************************
'4/16/21 InputFile2              << Comment in Procedure
' Final System Test Snippet
' Example Procedure Semi-Auto Test
' Open Comm Port for Testing
COM                        << Command to UUT
' Functional Testing
' Using the Semi-Automatic
' Test Program, Perform the
' Testing up thru Calibration
' Test
```

```
'******************************
'******************************
' 2.5 mm Cable Test
'****************************
' Delay 1 mS Test
'****************************
DLY 1                      << Command to UUT
DELAY 1 mS

*************************************************
   >>>>>>>>>>>>>>>> Test OK <<<<<<<<<<<<<<<<     *
*************************************************
'****************************
' Read Empty File Profile
'****************************
READ 1                           << Command to UUT
Number or Command Invalid
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
'****************************
' Read Test File Profile
'****************************
READ 13                          << Command to UUT
/*  Empty Location ......................................
*************************************************
   >>>>>>>>>>>>>>>> Test OK <<<<<<<<<<<<<<<<     *
*************************************************
'****************************
' Read Help File Profile
'****************************
READ 16                          << Command to UUT
Number or Command Invalid
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
'****************************
' Perform Relay 1 Test
' SW Relay Closed
RLY1 1                           << Command to UUT
/*A864 12/16/17 06:57:42 F DOC21341-090_rev-A APD.idf 12/15/
*************************************************
   >>>>>>>>>>>>>>>> Test OK <<<<<<<<<<<<<<<<     *
*************************************************
'****************************
' Perform Relay 1 Test
' SW Relay Open
RLY1 0                           << Command to UUT
/*
Help File - Demo Model II Tester - Firmware Revision 0.42
*************************************************
   >>>>>>>>>>>>>>>> Test OK <<<<<<<<<<<<<<<<     *
*************************************************
'******************************
'******************************
'
'CODE DELETED                     << Actual procedure code removed
'
'******************************
```

```
'******************************
' Set WRK Gain to Mode 2
' Observe LED I3 = ON
'****************************
RLY2 1                          << Command to UUT
Relay 1 = 1

*************************************************
   >>>>>>>>>>>>>>>> Test OK <<<<<<<<<<<<<<<<     *
*************************************************
'****************************
' WRK Verify Zero
' 0 nA +/-5 nA
' WRK 0
WRK 0                       << Command to UUT
Relay 1 = 0                 << Response From  UUT

*************************************************
   >>>>>>>>>>>>>>>> Test OK <<<<<<<<<<<<<<<<     *
*************************************************
'
'MORE CODE DELETED               << Actual procedure code removed
'
'******************************
'******************************
' Perform Square Wave Test
' Measure Voltage at TP1
' Enter Command Mode
' Enter SQWV 1 - Start Square Wave
' Hit Enter to Stop Square Wave
SQWV                        << Command to UUT
DAC_UB = 0, DAC_LB = 0      << Response From UUT
*************************************************
   >>>>>>>>>>>>>>>> Test OK <<<<<<<<<<<<<<<<     *
*************************************************
C To Continue, press Enter
S
Number or Command Invalid
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
   >>>>>>>>>>>>>>>> Test Failed <<<<<<<<<<<<<<<<
'****************************
' Perform Real Meter Test
' Using Term Emulator After
' Completing Square Wave Test
'****************************
' After Last Demo Unit is Tested
' Quit this Program
' ***************************
' ***************************
/END TEST THIS UUT              << Command to End Test of this UUT
*****************************************
*   >>>> Finished Testing UUT # 5 <<<<<   *
*****************************************
  Test Person's Name is: CAA

***************************************** << Status of Test Session
*             End of Testing Session        *
*        04/22/2021          13:40:47       *
```

```
*      Number Failed 3 out of 1 tested      *
*****************************************
```

Expanding the Test System I/O

The system can have further expandable Input or Output Sources or Loads by adding an expansion input or output box. These signals are connected to another pair of switches located on the main switch box. These switches permit signals up to about 100KHz bandwidth to be input to or read from the large switch array. These I/O devices might be connected to another expansion switch box that accomplishes the additional switching required to select multiple added inputs and outputs beyond those provided.

Summary of this Article

An inexpensive and practical Semi-Automatic Tester can be created to meet the needs of many PC-based testing requirements. Such a tester can be produced at a much lower cost and made more quickly available than constructing an entirely new single-purpose custom tester.

To learn more about the Custom Tester development provided by Angotti Product Development, check out our webpage at https://www.angotti.com/Test%20Engineering%20Menu.html or send an email to carl@angotti.com or phone 408-462-2189.