Identifying Legacy Re-Host Conversion Candidates: Keeping Current with Technology

Larry V. Kirkland
WesTest Engineering
810 Shepard Lane
Farmington, Utah 84025

Abstract - **How to identify when a static pattern digital test program should be re-hosted as a dynamic pattern set on the state-of-the-art ATE is complicated. However, there are critical issues that should be evaluated when making this decision. Issues include chip models, timing parameters, drive strength, logic levels, circuit complexity, I/O pins, automated diagnostics, portability, reuse, speed, optimal circuit coverage, schematics and data availability, etc.**

I. INTRODUCTION

The Test Program Set (TPS) re-host design, development, manufacture, integration and debug for a Line Replaceable Unit (LRU) and Shop Replaceable Unit (SRU) includes the analysis, assessment and improvement of the legacy test program, component fault coverage and diagnostic isolation capabilities. Also, a TPS re-host includes an analysis of Cannot Duplicate (CND) and Re-test OK (RTOK) occurrences. This comprises the quality of the re-hosted TPS. Often, a re-hosted TPS becomes a complete re-development or new TPS.

With the advent of robust Digital Simulators and state-of-the-art dynamic hardware is it feasible or even the right thing to re-host a static pattern set from a legacy ATE to a static pattern set on a state-of-the-art dynamic ATE? Generally, a static pattern set requires digital circuitry in the legacy ATE Interface Test Adapter (ITA) to match the timing parameters required for the UUT. Preferably, it is not wise to duplicate the legacy ITA hardware especially if the contract calls for straight-wire ITAs (if possible).

Converting a static test to a dynamic test requires development of a timing set and component models. This is intensive time consuming engineering work which could require additional up-front costs. However, this action will probably result in a reduction in overall life-cycle costs. Also, this makes the TPS more portable from tester to tester. Even if it means shutting down or adding a ground clip to an oscillator or other driving device, it becomes more cost effective and makes the TPS more portable and robust.

This paper will cover a conversion process for static to dynamic. The paper will show how the ITA components are deleted and what to look for in the legacy ITA components for signal timing and routing. A conversion sequence or check list will be included so as to provide a step by step sequence of events and engineering analysis. Also, this paper will provide a straight transfer routine using existing code to a state-of-the-art platform.

II. DERIVING THE PATTERNS AND TIMING

As a general rule, when we are re-hosting a digital legacy TPS we do not want ITA components. Often in a legacy ATE TPS, ITA components were added to setup timing characteristics for stimulus patterns. Also, response patterns can be processed through ITA components in legacy TPSs. ITA components for digital applications have become an industry NO-NO. The additional cost, due to component heavy ITAs, of an extensive ITA self test and the ITA reliability are prohibitive. Also, ITA components make future rehosts and software support major issues.

Patterns and timing are obtained by reviewing both the legacy ITA hardware and the legacy digital stimulus. Patterns are a combination of the actual digital stimulus from the legacy instruments and the fan-out from ITA components. You will need to find the actual source of all digital signals supplied to the UUT and received from the UUT in the legacy TPS. Remember, perhaps only certain stimulus and sensor signals are active for any given set of digital tests (HiZ is a real consideration).

III. LEGACY TPS ITA TIMING

Deleting digital ITA components requires an analysis of the delay or timing characteristics of the legacy ITA components. The delay for a single gate is straightforward. The delay for a gate is derived from data components data charts. Often it is logical to use the worst propagation delay for a chip.

The actual digital signal of a legacy ATE to an ITA component set to the UUT pin is the route of electrical nodes from a given input to a particular output. This is known as the legacy ITA signal delay or timing path. There may be none, one, or more delay paths between an input and to the output. Also, the signal logic type (Hi or Lo) may change. Figure 1 is an example of legacy ITA components.

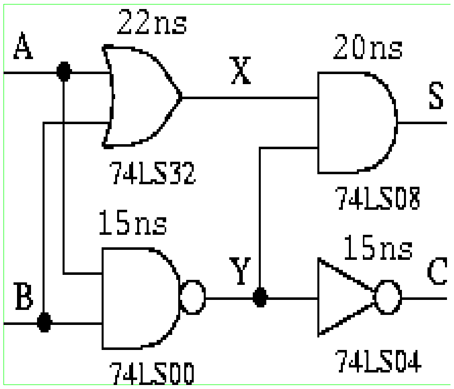


Figure 1. Example of legacy ITA components

If all the gates have approximately the same worst case delay, pick the path with the most number of gates[1]. Otherwise, add up the delays for each path and pick the path with the largest sum[1].

A to S - two paths[1]
22ns + 20ns = 42ns <- largest
15ns + 20ns = 35ns

A to C - one path[1]
15ns + 15ns = 30ns

B to S - two paths[1]
22ns + 20ns = 42ns <- largest
o 15ns + 20ns = 35ns

B to S - one path[1]
15ns + 15ns = 30ns

Summarize the delays in a table[1]. Table 1 shows the gate delay example.

	S	C
A	42ns	30ns
B	42ns	30ns

Table 1. Gate Delay Example

When creating a new ITA with straight wires only, the delays shown above will need to be added to the pattern set for the rehosted dynamic TPS.

IV. COMPONENT MODELING

Component models (usually inserted in a library) consist of one or more elements which describe the logical operation and timing of the components. Most models are described by three elements[2]:

1. The structural interconnection description for the component.
2. A listing of the functional timing specification for the component I/O pins in terms of the values assigned in the TVA element. Operational timing is specified for input and delays are specified for output pins.
3. The characteristics typically exhibited by components of the indicated technology (TTL, CMOS, ECL, …) Signal strength specifications from either primitives or component outputs could also be included in this element.

V. SOURCES OF COMPONENT MODELING

There are five sources of component models[2]:

1. You can purchase Teradyne Models.
2. You can use an editor to manually create structural models.
3. You can create components using computer-aided modeling programs.
4. You can use the LASAR Software Behavioral Language to describe the behavior of a component without giving its structural description. This method generates only the logical description equivalent to a structural LAS element for the component. You must add the TVA and TFU elements.

VI. OVERALL DEVELOPMENT

You can redesign and re-determine legacy software patterns by evaluating the legacy pattern set. Often the legacy static pattern set is setup properly for signal occurrence. You will need to setup the software controlled timing characteristics based on your analysis of the legacy ITA hardware.

Step by step:

1. Evaluate legacy timing sequences for each stimulus signal.
2. Determine if legacy pattern sequence is properly setup for the timing sequence including ITA hardware analysis.
3. Duplicate timing sequence in software.
4. Apply patterns using a simulator and review any hazards that occur.
5. Note that hazards are likely to show-up since the legacy timing was performed manually with out the aid of an advanced simulator.
6. Correct hazards caused by timing errors.

7. Add patterns if necessary to achieve proper detect (usually 95%).
8. Remember, you will need to verify testing functionality on the ATE during integration.

VII. ANOTHER OPTION

If the legacy ITA has no components or negligible number of components and has a static pattern set but you do not have all the component models for the UUT, one option is:

1. Identify I/O pins
2. Duplicate the pattern set with some timing adjustments using Teradyne's Soft Front Panel
3. Set outputs based on test results
4. Associate chips for back trace probe
5. Run a specific number of patterns (duplicate legacy sequences for fault determination) and test for a failure

You can utilize a partial DTB program:

> The Partial DTB program is used to execute part of a .dtb pattern set. We do not need to add any Halts on the patterns we want to stop anymore. You just need to send thru Atlas the start and stop patterns of the .dtb. Keep in mind that the .dtb pattern starts at pattern 0, so for example, if you want to execute from the beginning of your .dtb, then your start pattern (sent from Atlas to this program) should be 0 (just like the .dtb file).

6. When a failure occurs at a specific pattern you can perform diagnostics by either back trace probing or simply calling out an R/R (see figure 2).

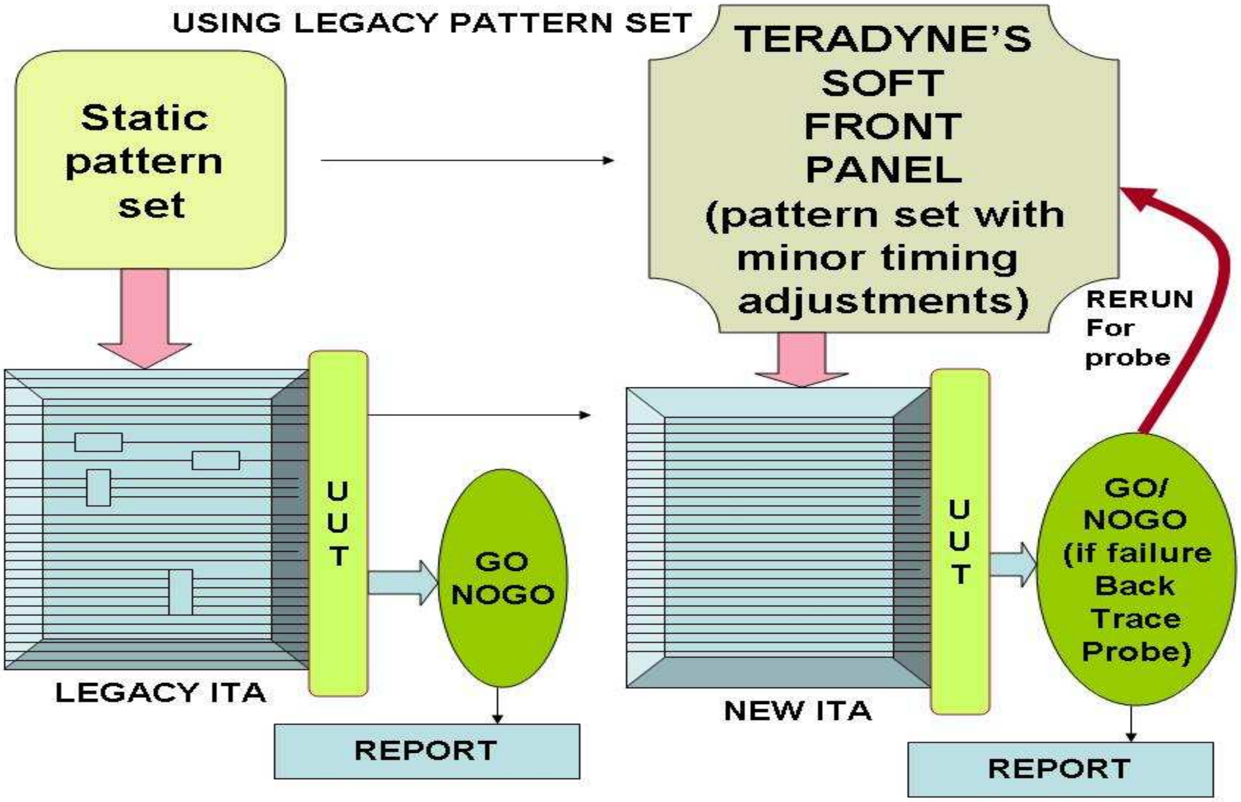
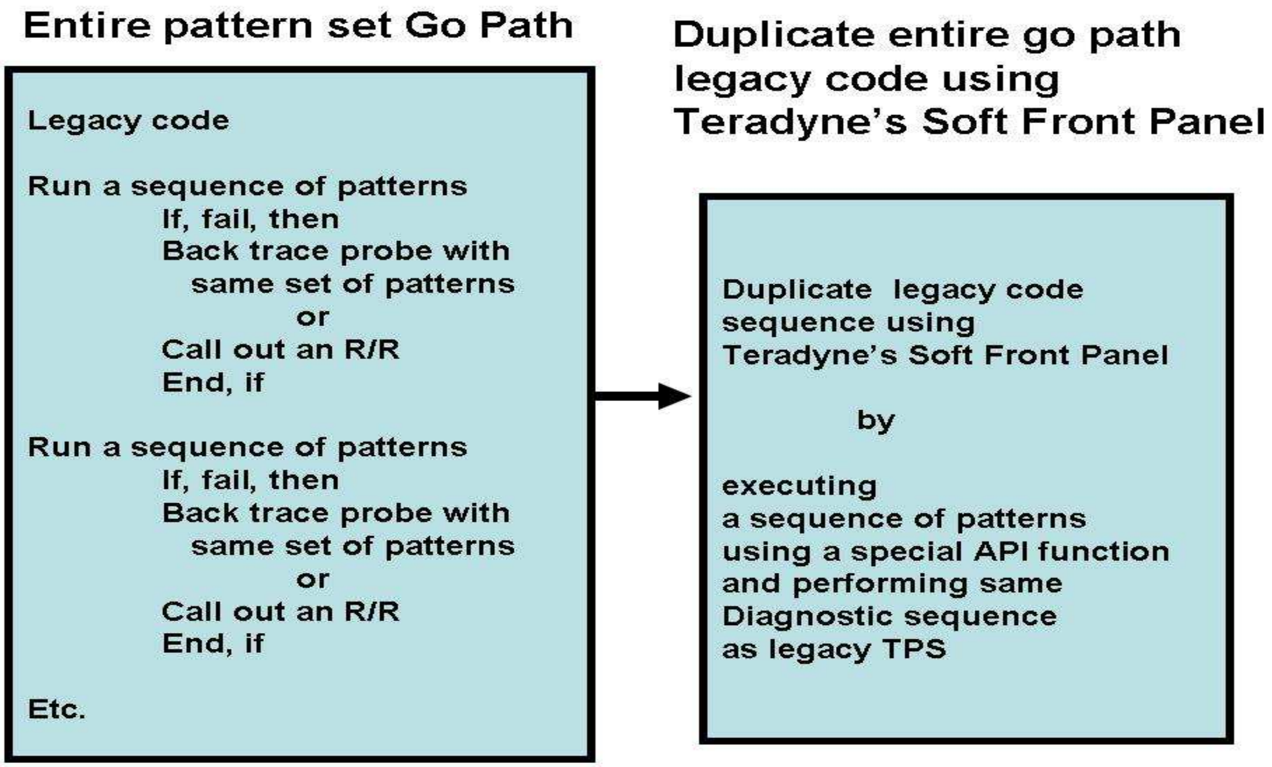

USING LEGACY PATTERN SET
Static pattern set
TERADYNE'S SOFT FRONT PANEL (pattern set with minor timing adjustments)
RERUN For probe
UUT
GO NOGO
LEGACY ITA
REPORT
UUT
GO/ NOGO (if failure Back Trace Probe)
NEW ITA
REPORT

Figure 2. Using legacy pattern set

Essentially, you can duplicate the entire legacy go-path sequence on one Teradyne Soft Front Panel dtb. File. Then, precisely duplicate the diagnostic sequence by exactly running a specific number of patterns (see figure 3) and if, fail, then run the same diagnostic sequence as the legacy TPS. The is accomplished by following the figure 3 for the "Entire Pattern set Go Path" but using the soft front panel with specific code sequences. The method is preferable over creating many dtb files.

Creating a large number of dtb files can introduce many unforeseen problems. These problems can cause testing glitches, erroneous failures, equipment timing setup problems, and complete Go-path flow.



Entire pattern set Go Path

```
Legacy code

Run a sequence of patterns
        If, fail, then
        Back trace probe with
          same set of patterns
                or
        Call out an R/R
        End, if

Run a sequence of patterns
        If, fail, then
        Back trace probe with
          same set of patterns
                or
        Call out an R/R
        End, if

Etc.
```

Duplicate entire go path legacy code using Teradyne's Soft Front Panel

Duplicate legacy code sequence using Teradyne's Soft Front Panel

by

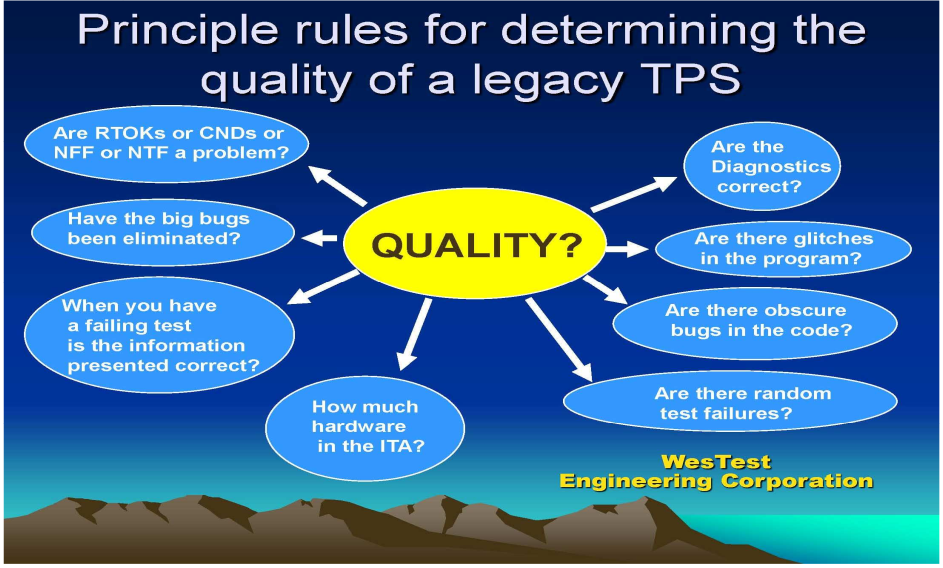
executing
a sequence of patterns
using a special API function
and performing same
Diagnostic sequence
as legacy TPS

Figure 3. Duplicate legacy TPS

This option is applicable under the following conditions:

1. All the component models for the UUT are not obtainable.
2. The legacy TPS has no RTOK problems.
3. There are no glitches in the legacy TPS.
4. The user prefers the existing TPS because the diagnostics works great.
5. Have the big bugs been eliminated.

Figure 3 depicts the quality determination characteristics of a legacy TPS. By experience, I know some users would rather go onto another job rather than anyone changing the TPS they know so well.



Principle rules for determining the quality of a legacy TPS

QUALITY?

Are RTOKs or CNDs or NFF or NTF a problem?

Have the big bugs been eliminated?

When you have a failing test is the information presented correct?

How much hardware in the ITA?

Are the Diagnostics correct?

Are there glitches in the program?

Are there obscure bugs in the code?

Are there random test failures?

WesTest Engineering Corporation

Figure 3. Quality of a Legacy TPS[3]

VIII. CONCLUSION

It is always best when rehosting a digital TPS to have a no-component straight-wire ITA. This has significant long term support savings and sets up ease of future rehost as test technology evolves. It is preferable to use dynamic timing methods whenever a digital TPS is rehosted.

The deciding TPS factor for rehosting a legacy TPS must always be the quality of the TPS coupled with long term costs and life-cycle support.

There are things to realize:

- Many factors are applicable
- Develop the best possible TPS (actual TPS usage is critical)
- Some legacy TPS developers were brilliant
- If the legacy TPS is faulty in anyway, then think re-development
- Some legacy TPSs are unacceptable
- Re-hosting could mean complete re-development
- Diagnostics is very, very, very important: too many probes and/or high callouts can be a reason to retain the legacy TPS as is
- An exact duplication of a legacy TPS is possible on state-of-the-art equipment
- Don't forget to ask the user some critical questions about the legacy TPS
- Review legacy TPS problems by looking at previous reports like a Material Improvement Report and the Material Improvement Project with final analysis
- Review test sequence flow, some tests might be mixed up

It is best to use state-of-the-art testing methods even if there is additional cost upfront. Also, with the scoring method of modern digital simulators, we can improve the TPS detection quality considerably.

However, after all is said and done, if a legacy TPS is performing great and there are no RTOK or testing problems a good second look should be performed to determine the best approach to rehosting. The ultimate rehost method should be to create a highly portable TPS with little or no ITA components that does the best job.

An excellent Test Strategy analysis should include evaluating existing legacy code. At times it might seem legacy code does not seem to properly examine Input/Output pin coverage, component failure mode coverage and ambiguity group size. However, depending on the TPS, legacy TPS test strategies were often based on factors unbeknownst to the rehost engineer.

References

[1] Jonathan Hill *jmhill* at *hartford* dot *edu*

[2] TERADYNE, Defense and Aerospace Division

[3] KIRKLAND, Larry, AUTOTESTCON 2009