Test Fixture and Program Development

Terms and Conditions

1. CheckSum, LLC does not accept liability for damage to provided samples. Customer allows that boards sent to CheckSum as samples will not be returned as shippable product for end users.
2. Work requested by Customer, beyond the scope of this document, may result in additional costs.
3. Not all customization is always possible. For example, tooling pins may be close to components making the hollowed out pressure rod over the top of the tooling pin untenable.
4. CheckSum only warrants fixtures where the revision of the data matches the revision of the board samples supplied.

Test Program Coverage

This section describes the process CheckSum uses to develop each ICT/MDA test program.

Although each assembly is unique and the test coverage results may differ, we attempt to provide a test program that maximizes test coverage at a reasonable cost.

For this explanation, access to all nets is assumed. Limited access will result in limited coverage in normal situations.

CheckSum asks for known-good boards to aid in our test development. Boards that have errors will result in longer development time and may cause errors such as false failures.

In order to provide cost-effective, ICT/MDA programming and fixturing, faults are not injected into the sample assemblies. To ensure maximum test coverage, you may wish to manually inject faults (e.g., remove parts or insert parts backwards) and ensure that these faults are found.

For each assembly, CheckSum will perform the following:

Continuity Test

CheckSum will probe each net once, unless otherwise specified by the customer. The continuity test will check for opens and shorts on the sample provided. The test system will “learn” the continuity map from the provided “good” sample assemblies. If the sample has errors, the test may end up with those same errors. To detect a missing connector with the continuity test, a test probe needs to contact the connector pins.

In-Circuit Component Test

For each component (resistor, capacitor, diode, zener, LED, IC, inductor, transformer, relay, switch, jumper, etc) we will create one or more test steps:

First, the optimum measurement range and measurement type (constant-current or constant-voltage) is selected. If the measured value is not within the expected tolerance, we will attempt to change the method of test to better isolate the component. For example, we will add guard points as needed and/or increase the measurement setting time.

For resistors in parallel, they are tested in parallel. The test step title indicates parallel resistors.

For capacitors in parallel, they are tested in parallel. The test step title indicates parallel capacitors.

The variation of the actual capacitor values tends to not allow failure discrimination. When components are measured in parallel, some missing components may not be detected. For example, VCC to GND typically has a larger capacitor in parallel with a number of smaller capacitors. If one or more of the smaller capacitors is missing, the system is likely not to detect it since its value is small compared to the acceptable range of values of the large capacitor.

If a low impedance path is in parallel with a high-impedance component, the high-impedance component may not be testable. An example is a resistor in parallel with an inductor or transformer winding.

Transistors are tested with diode tests and Beta tests (TR-8 and TR-10 versions only). For a standard (NPN or PNP) 3-leg transistor device, we will setup (2) diode tests from Emitter-Base and Collector-Base. When a self-learn occurs, the System monitors the collector voltage and determines the voltage threshold point at which the transistor turns-on. The beta tests are not always repeatable and will be removed if we can't produce consistent results.

Integrated-Circuit (IC) Test

The IC test looks for diode junctions, typically found between each IC I/O pin and the power and ground supply pins. This test works in a similar fashion to the continuity test where the sample is "learned" for all the nets on the assembly. Subsequent tests are then compared to the learned map.

This test can catch some parts incorrectly installed, but not necessarily every part.

Validation

The last step to validate the test program is to use the built-in statistics (SPC). Using as many sample assemblies as are provided, multiple runs are tested and the results are examined. The goal is to insure the final measurements are centered in the limits and the Cpk value is greater than 3.

As a final quality assurance check, a 2nd Test Engineer reviews the statistical (SPC) results, the test fixture and test program, and documentation package.

Additional testing options

CheckSum test equipment has additional capabilities beyond the basic ICT steps listed above. The capabilities include TestJet, Boundary Scan, on-board part programming, power-on and functional tests. All of these tests require additional modules that may or may not already be part of your test system.

For power-on and functional tests CheckSum prefers to work from a customer supplied test specification.

Contact

If you are interested in expanding your CheckSum system test capabilities, have any questions pertaining to this document, or the development of your test fixture, please contact:

Fixture Project Coordinator
CheckSum, LLC
(360) 435-5510
Email Fixture.Project.Coordinator@checksum.com