

## Übersicht

Mit dem XJDeveloper lassen sich an ihrer Schaltung schnell und einfach Tests aufzubauen und durchführen. Der „Build-In Connection Test“, in Verbindung mit XJEase für die Nicht-JTAG Bauteile, prüft die Baugruppe nach Kurzschlüssen und offenen Verbindungen. Der XJDeveloper erlaubt ihnen in einfacher Weise die Test anzupassen und zu modifizieren. Die eingebaute DFT Analyse gibt ihnen einen Überblick über die Baugruppen Testabdeckung.

Mit XJDeveloper lassen sich sowohl JTAG Bauteile (z.B. CPLD's, FPGA's) als auch Nicht-JTAG Bauteile (z.B. EEPROM's, Flash) programmieren.

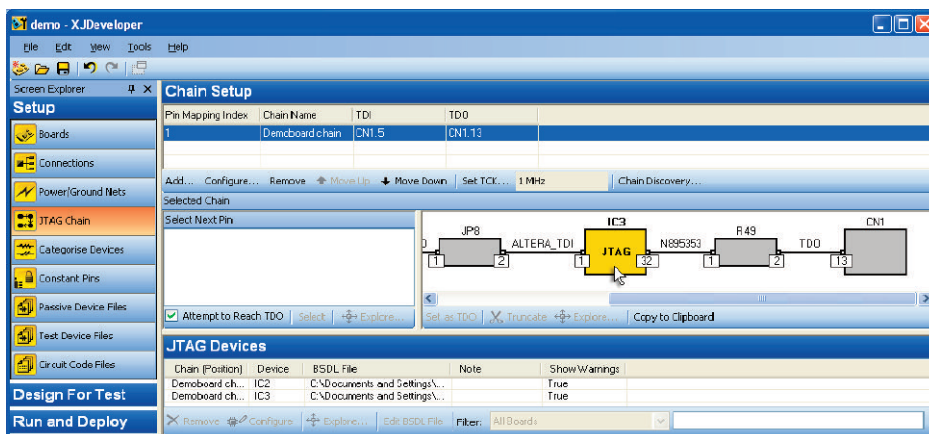
### Test ihrer Baugruppe

XJDeveloper und XJEase verwenden die Informationen aus Ihren Netzlisten und aus den BSDL(Boundary Scan Description Language)-Dateien, um die Verbindungen auf den zu testenden Baugruppen zu überprüfen.

XJTAG unterstützt derzeit über 50 verschiedene Netzlistenformate. Die BSDL-Dateien erhalten Sie in der Regel kostenlos auf der Webseite der Bauteilehersteller.

### Nicht-JTAG Testen

In XJDeveloper können Sie sehr einfach über die Bausteine in der JTAG-Kette auch die Verbindungen der Komponenten ohne JTAG-Interface testen. Wenn Sie z.B. Testdaten in einen Speicherbaustein schreiben und wieder zurück lesen, können Sie feststellen, ob die Daten- und Adressleitungen frei von Kurzschlüssen oder Unterbrechungen sind. Sie können auch wesentlich komplexere Testabläufe realisieren, wie z.B. die Übertragung von Ethernet-Paketen – und das ohne vorher den Prozessor booten zu müssen.



Konfigurieren Sie Ihr Projekt mit dem XJDeveloper

### XJEase

XJEase ist eine Hochsprache für die Programmierung kompletter JTAG-Testlösungen mit der vollen Funktionalität, Flexibilität und Kontrolle, die Sie benötigen. In den meisten Fällen finden Sie den XJEase code für ihre Nicht-JTAG Bauteile auf der XJTAG Webseite zum Download. Aber auch das schnelle Schreiben neuer Tests oder kundenspezifische Anpassungen ist einfach und ohne spezielle Programmierkenntnisse machbar.

Ein möglichst einfacher Zugriff auf die Hardware war das wichtigste Entwicklungsziel von XJEase. Legen Sie einfach fest, welche Anschlüsse der Non-JTAG-Bausteine angesteuert und welche gelesen werden sollen. Sie müssen sich nicht darum kümmern, über welche JTAG-Bausteine Sie diese Pins erreichen oder wie der passende JTAG-Befehl dazu lautet. Mit XJEase haben Sie die vollständige Kontrolle über den Ablauf Ihrer Tests.

## Die wichtigsten Vorteile

- Reduzierter Zeitaufwand bei der Fehlersuche auf dem Board durch exakte Fehlereingrenzung
- Verkürztes Time-to-Market und reduziertes Projektrisiko auf Grund einer frühzeitigen Design-Verifizierung
- Die Wiederverwendbarkeit der Tests für Entwicklung / Prototyping bis hin zur Fertigung und Wartung verkürzt die Test-Entwicklungszeiten drastisch
- Weitere Zeiteinsparungen durch die Wiederverwendung von Tests zwischen unterschiedlichen Projekten

## Schnelle Test-Entwicklung

Die Tests für einen Baustein ohne JTAG-Schnittstelle werden für diese Komponente unabhängig vom Rest der Schaltung geschrieben. Dadurch sind Sie in der Lage, den Test immer wieder zu verwenden, wenn der Baustein in einem zukünftigen Design zum Einsatz kommt. So sparen Sie sehr viel Zeit und Geld.

XJEase bietet Ihnen alle Vorzüge, die Sie von der Hochsprachen-Programmierung gewohnt sind: Variablen, Schleifen, bedingte Ausführung, Funktionsaufrufe und vieles mehr. Sie können Ihr Board in Echtzeit kontrollieren und nicht nur Werte setzen und prüfen. Der XJDeveloper „Build-In test editor“ macht das Schreiben der Tests einfach und schnell.

## Kostenlose Bibliothek

Um Bausteine ohne JTAG-Schnittstelle zu testen, steht Ihnen auf [www.xjtag.com](http://www.xjtag.com) eine umfangreiche Bibliothek von XJEase-Chip-Files zum kostenlosen Download zur Verfügung. Selbst wenn Sie XJTAG das allererste Mal einsetzen, ist es möglich, einen vollständigen funktionierenden Test ohne jede zusätzliche Programmierung zu realisieren.

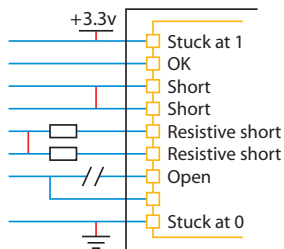
Wenn Sie die Datei für genau Ihren Baustein nicht finden sollten, laden Sie einfach ein ähnliches File herunter und

passen es in wenigen Schritten an. Außerdem stehen wir Ihnen beratend und unterstützend zur Seite, damit der Test genau Ihren Anforderungen entspricht.

## Verbindungstest

XJDeveloper verfügt über einen integrierten Verbindungstest für alle Anschlüsse der JTAG-Bausteine. Abhängig davon, in welchem Maß der Zugriff über den Boundary Scan möglich ist, überprüft der Verbindungstest die Leiterbahnen auf Kurzschlüsse und Unterbrechungen – auch Kurzschlüsse zur Versorgung oder Masse sowie widerstandsbehaftete Kurzschlüsse und Kurzschlüsse über einen Inverter. Auch Pull-up- und Pull-down-Widerstände werden in den automatischen Verbindungstest einbezogen.

Nachdem ein Fehler erkannt wurde, führt der Verbindungstest abhängig vom Aufbau des Boards weitere Analysen durch, um den Fehlerort weiter einzugrenzen. Viele unserer Kunden haben Fehler auf Baugruppen gefunden, die vorher als fehlerfrei eingestuft waren.



Mit dem XJTAG Verbindungstest ermitteln Sie die Mehrzahl aller Fehler

## Flash Programmierung

Auf der XJTAG-Website steht Ihnen eine große Anzahl Device Files für Flash-Bausteine zum Download bereit. Darunter auch Funktionen für die Programmierung der Bausteine mit jedem beliebigen Image. Diese Dateien nutzen die Programmieralgorithmen, die in den Datenblättern angegeben sind, um beliebige Bausteine ohne JTAG-Unterstützung testen zu können. Es ist nicht notwendig, die Verbindungen zwischen dem Flash und dem JTAG-Baustein aufzulisten. Dazu verwendet

XJDeveloper die Netzliste und führt diese Aufgabe automatisch durch. Das darunter liegende System generiert die erforderlichen Vektoren, um den Flash-Baustein zu programmieren.

## Analyse der Testabdeckung

Sobald Sie über einen ersten Schaltplanentwurf verfügen, können Sie analysieren, wie viel von Ihrem Board durch den XJDeveloper-Test abgedeckt wird. So können Sie sehen, wo Sie mit zusätzlichen Verbindungen die Abdeckung verbessern können. Die Testabdeckung wird automatisch aus der Kombination von XJEase-Verbindungstest und dem zusätzlichen Test der Bausteine ohne JTAG-Unterstützung berechnet. Um die volle Funktionalität von XJDeveloper ausnutzen zu können, hat XJTAG ein DFT-Handbuch (Design for Testability) als Referenz-Leitfaden erstellt.

## Programmierung von CPLD/FPGA

Viele JTAG-Bausteine wie CPLD und FPGA können direkt über Dateien im Format STAPL / JAM und SVF programmiert werden, die mit den Werkzeugen der Chiphersteller erzeugt wurden. Die Programmierung wird in einem Stand-alone-Prozess oder integriert in XJDeveloper mit einer einzigen Zeile Code vorgenommen.

## Integration

Das COM-Interface ermöglicht die XJTAG Test-Integration mit Testumgebungen wie NI LabVIEW oder andere proprietäre Anwendungen, die in Programmiersprachen wie Visual Basic® und C#® geschrieben sind.

## Kommt mit XJRunner

Das erlaubt ihnen abschließende Entwicklungstests bevor die Daten an die Fertigung übergeben werden.

Im XJRunner Datenblatt finden Sie weitere Informationen

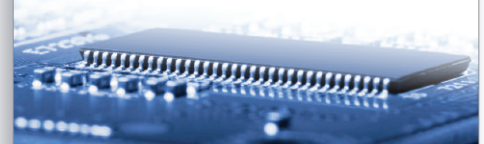
## Eigenschaften

- Analyse der Testabdeckung vor der Entwicklung der Leiterplatte
- Integrierter, adaptiver Verbindungstest
- Kostenlose Online Bibliothek von Standardbauteilen
- Programmierung von CPLD, FPGA, Flash-Bausteinen usw.
- Erweiterte Testfunktionen — z.B. Ethernet Loopback
- Integration in andere Applikationen, um eine vollständige Testumgebung zu entwickeln
- Unterstützung von Netzlisten im Format EDIF 2.0.0, RINF, Protel, PADS\_PCB, ALLEGRO und viele andere
- Unterstützt IEEE 1149.1 und 1149.6-Testfunktionen

## XJTAG bietet mehr...

Beim Kauf von XJDeveloper erhalten Sie alle genannten Funktionen. Außerdem im Preis:

- XJRunner — die spezielle Laufzeitumgebung zum Test Ihrer Schaltung
- XJDebug — XJEase code ausführen von der Befehlszeile
- XJLink — das USB 2.0 zu JTAG Interface verbindet ihren PC mit der Schaltung die getestet werden soll
- Eine Floating Lizenz, die sich im XJLink befindet. Sie können die Software auf beliebig vielen PCs installieren
- Demo hardware
- Umfangreiches Tutorial



Vertriebspartner / Technologiepartner

[www.xjtag.com/partners](http://www.xjtag.com/partners)