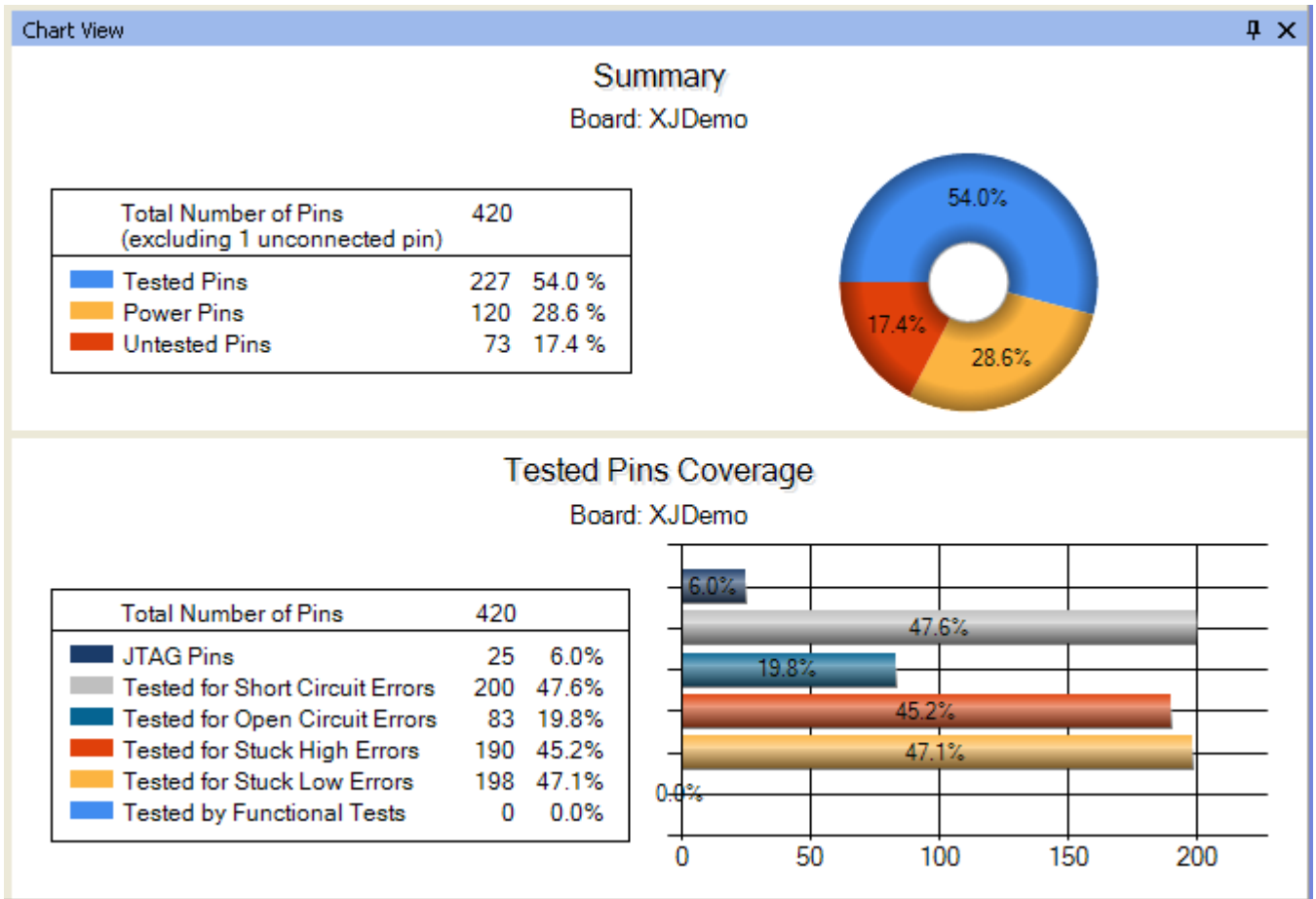


JTAG バウンダリスキャンテストのカバレッジを考察することは、基板テスト設計の大切な要素です。XJDeveloper を単独、あるいは複合的なテストソリューションの一部に使用する場合のどちらにも、Design For Test 解析結果から有益な情報を得ることができます。この例では XJTAG バウンダリスキャンテストシステムに付属の XJDemo ボードを使用しますので、事前に XJDeveloper and XJEase のチュートリアルを理解されることをお勧めします。

テストカバレッジ情報が設定されていない XJDemo ボード用の XJDeveloper プロジェクトがあります。
(C:\Program Files\XJTAG xxx(インストールディレクトリ)\Help\demoNoDFT.zip)

- demoNoDFT.zip を任意のディレクトリに展開する。
- そのディレクトリ内に、スタート→XJTAG 2.5→XJTAG shared Files を開くことで得られる、Demo Board v3\Testcode 内の ODB++ディレクトリをコピーする。
- XJDeveloper で、このディレクトリ内の demo.xjd を開く。

XJDeveloper の Design For Test の DFT Analysis 画面を選択すると、XJDemo ボードのテストカバレッジが以下のように表示されます。




このテストカバレッジは、XJDeveloper 内に設定された回路情報を基に XJTAG が自動的に生成した接続テストから導き出されています。テストされないピン (untested pins) が 73 で全体の 17.4%とレポートされており、これは殆どのボードにとって妥当な結果と言えますが、XJDemo ボード上で達成できるテストカバレッジはもっと高くなります。

接続テストのみならず、XJDemo ボード用のテストには JTAG 未対応デバイスの機能を活用するものがあります。これらは、デバイスの機能をテストするために設計されたものではなく、デバイスの機能を利用してデバイスがボードに正しく接続されていることを検証するためのものです。


XJDeveloper は、これらテストを実行することで達成される追加のテストカバレッジを、自動的に包括して導くことはできません。テストされる欠陥の種類とバスを XJDeveloper に設定する必要があります。これは、XJDeveloper の Test Device Files セクションにある各バスの Tested for value を設定することで行います。

これら Tested For values を修正する前に、このボードの主な部品のカバレッジを確認します。

 **DFT Analysis** 画面左の **SummaryStatistics** 内で **XJDemo** を展開させてから、右上の **View Mode** 枠内の **Filter** フィールドに **IC** と入力すると以下のようにフィルター表示されます。

Summary Statistics											
Element	JTAG	Short	Open	StuckH	StuckL	Functional	Tested	Power	Untested	Total	
[-] Demonstration hardware	25	200	83	190	198	0	227	120	73	420	
[-] XJDemo	25	200	83	190	198	0	227	120	73	420	
+ IC1	4	35	20	32	35	0	39	6	1	46	
+ IC2	4	30	22	29	29	0	35	9	3	47	
+ IC3	0	26	0	24	26	0	26	9	1	36	
+ IC4	0	18	17	18	18	0	18	2	0	20	
+ IC5	0	26	0	24	26	0	26	2	0	28	
+ IC6	0	1	0	1	1	0	1	6	9	16	
+ IC7	0	1	0	1	1	0	1	2	1	4	
+ IC8	0	6	6	6	6	0	6	2	6	14	
+ IC9	0	2	2	2	2	0	2	2	0	4	

IC7 (IIC EEPROM) を例にとると、このデバイス用に実装されたテストは IIC インターフェース (SDA バスと SCL バス) の接続を検証します。どちらかのバスが開放、電源にスタック、グランドにスタックの状況にあれば、テストは失敗します。また、2つのバスが互いに短絡していてもテストは失敗します。しかしながら、接続テストで得られるような、ボード上の残りの部分に対する広範な短絡エラーのカバレッジは得られません。このテストで得られるカバレッジを XJDeveloper に反映させるには、テストの実装に使用している Test Device File を修正する必要があります。

- 左端 Screen Explorer 内の Setup の下にある  Test Device Files 画面選択ボタンをクリック。
- ファイル 24XXX.xje をクリック。
- Busses 表示内で SDA バスをクリック。
- この SDA 行の Tested For 列の下にある None をダブルクリック。
- 表示されるオプションから Opens、Stuck High、Stuck Low を選択。
- Busses 表示内の Tested For 以外の部分をクリックし、これら設定を反映させる。
- SCL バスに対しても同様の処理を行います。
- Test Device Files 表示下部にある Save をクリックします。

Bus Name	Pins	Disable Value	Tested For
SCL	6	None	<input checked="" type="checkbox"/> Opens
SDA	5	Input	<input type="checkbox"/> Shorts <input checked="" type="checkbox"/> Stuck High <input checked="" type="checkbox"/> Stuck Low <input type="checkbox"/> Functionally Tested

Create Bus... Edit Bus... Delete

他の部品の Tested For セクションも、以下のように設定しましょう。

SRAM__TSOP28.xje

Bus Name	Pins	Disable Value	Tested For
ADDRESS	8, 5, 9, 2, 28, 3, 4, 10, 11, 12, 13, 14...	None	Opens, Shorts, Stuck-1, Stuck-0
DATA	26, 25, 24, 23, 22, 20, 19, 18	None	Opens, Shorts, Stuck-1, Stuck-0
nCE	27	High	Opens, Stuck-1, Stuck-0
nOE	1	High	Opens, Stuck-1, Stuck-0
nWE	6	None	Opens, Stuck-1, Stuck-0

Create Bus... Edit Bus... Delete

ADS7830.xje

Bus Name	Pins	Disable Value	Tested For
ADC_CH	8, 7, 6, 5, 4, 3, 2, 1	None	None
GND	9, 11	None	None
I2C_ADDR	13, 12	None	None
P3V3	16	None	None
SCL	14	None	Opens, Stuck-1, Stuck-0
SDA	15	Input	Opens, Stuck-1, Stuck-0
VREF	10	None	None

Create Bus... Edit Bus... Delete

LED.xje

Bus Name	Pins	Disable Value	Tested For
ON_OFF	2	None	Opens, Stuck-1, Stuck-0

Create Bus... Edit Bus... Delete

pushbutton.xje

Bus Name	Pins	Disable Value	Tested For
BUTTON_OUT	3	None	Opens, Stuck-1, Stuck-0

Create Bus... Edit Bus... Delete

Crystal.xje

Bus Name	Pins	Disable Value	Tested For
nOUTPUT	2	None	Opens, Stuck-1, Stuck-0
OUTPUT	1	None	Opens, Stuck-1, Stuck-0

Create Bus... Edit Bus... Delete

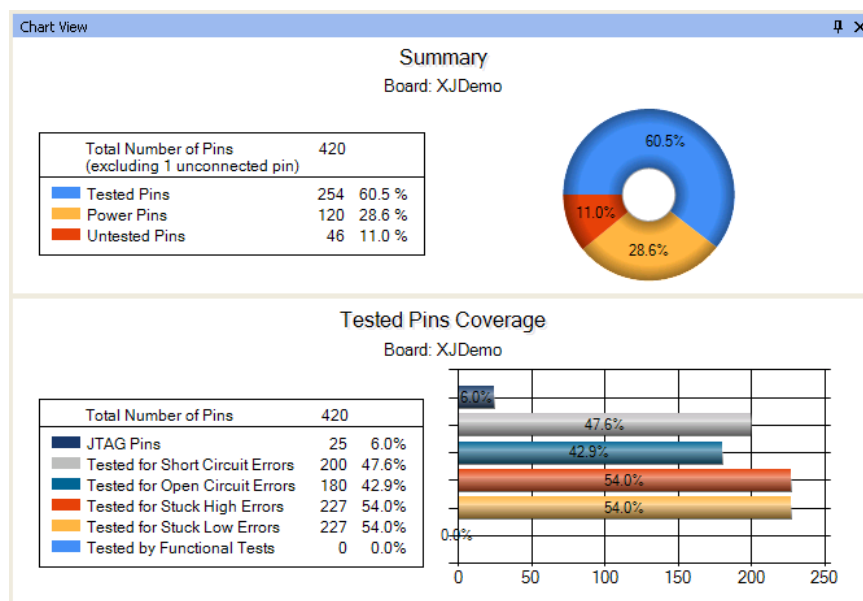
A29L400A.xje

Bus Name	Pins	Disable Value	Tested For
ADDR	3, 4, 5, 6, 7, 8, 18, 19, 20, 21, 22, 23...	None	Opens, Shorts, Stuck-1, Stuck-0
DATA	44, 42, 40, 38, 35, 33, 31, 29	None	Opens, Shorts, Stuck-1, Stuck-0
NBYTE	47	None	None
NCE	26	High	Opens, Stuck-1, Stuck-0
NDE	28	High	Opens, Stuck-1, Stuck-0
NRESET	12	None	None
NWE	11	None	Opens, Stuck-1, Stuck-0
RDY	15	None	None

Create Bus... Edit Bus... Delete


下図のように改めて **DFT** サマリを表示させると、デバイスファイルに対して追加されたテストカバレッジが反映されています。テストされないピンの数が46（11.0%）に減少したことが確認できます。そして開放エラーのカバレッジに着目すると、元のサマリでは83ピン（19.8%）のカバレッジが、テストデバイスファイルの **Tested For** に情報を追加したことで180ピン（42.9%）に増加しています。

開放エラーのテストカバレッジが大幅に向上していることがわかります。**JTAG** バウンダリスキャンの接続テストでは、短絡エラー検出には優れているものの開放エラー検出が難しいためです。接続テストで開放エラーを検証するためには、あるピンからネットを駆動した値を他のピンから読み取る必要があります。これは、ひとつのネット上に少なくとも2つの **JTAG** 対応ピンがあるという特別な状況が要求されます。それゆえ回路内の **JTAG** 未対応デバイスの機能を使ったテストを追加することで、開放エラーのテストカバレッジを大幅に向上させることができるのです。



次のステップではテストカバレッジの更なる向上を検討します。**XJDeveloper** の **DFT Analysis** セクションで、回路の **untested** セクションをハイライト表示します。




- **DFT Analysis** 画面の右上にある **View Mode** 枠下部の **Pin Filter** タブをクリック。
- **Show only pins that are not tested** ボタンをクリック。
- **DFT Analysis** 画面の左側の **Summary Statistics (filtered)** セクションで、**XJDemo** の左の+マークをクリック。

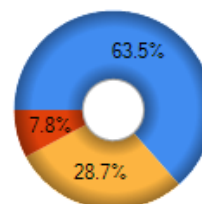
改めて  **DFT Analysis** 画面で IC6 を確認すると **Untested pins** の数が 8 から 4 に減りました。一方この **Functional Test** の影響は IC6 以外にも及んでいます。

●  **DFT Analysis** 画面の左側の **Summary Statistics(Filterd)** セクションで、XJDemo を選択します。

ボード上でテストされるピンのカバレッジを確認すると、**ADC Test** によるテストカバレッジは 13 ピンになっています。これには、**IC6** の 4 つのアナログ入力ピンと、これらに接続されているネット上の他のピンが含まれます。結果この **Functional Test** はテストされないピンを 33 ピン (7.9%) まで削減します。








Summary
Board: XJDemo

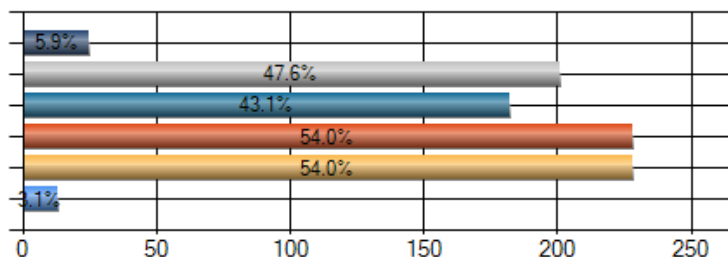
Total Number of Pins (excluding 1 unconnected pin)	422
 Tested Pins	268 63.5%
 Power Pins	121 28.7%
 Untested Pins	33 7.8%



Tested Pins Coverage

Board: XJDemo

Total Number of Pins	422
 JTAG Pins	25 5.9%
 Tested for Short Circuit Errors	201 47.6%
 Tested for Open Circuit Errors	182 43.1%
 Tested for Stuck High Errors	228 54.0%
 Tested for Stuck Low Errors	228 54.0%
 Tested by Functional Tests	13 3.1%
 ADC Test	13 3.1%



他にも、このプロジェクトに追加することでテストされないピンを 16 にまで減らすことができる **Functional Test** が、XJTAG 共有フォルダ内の別プロジェクトにあります。そして残る 16 ピンは ADC のテストされない 4 入力とこれらのピンが接続されているネット上のピンです。

これら残りにもテストカバレッジを得るには、これらアナログ入力をシミュレートする何らかの仕組みで ADC をチェックする必要があります。その一例として XJIO ボードを活用して XJDemo ボードのテストカバレッジを 100%にする方法の説明があります。(XJTAG Help→XJIO board→XJIO Board Version 2→Example を参照)