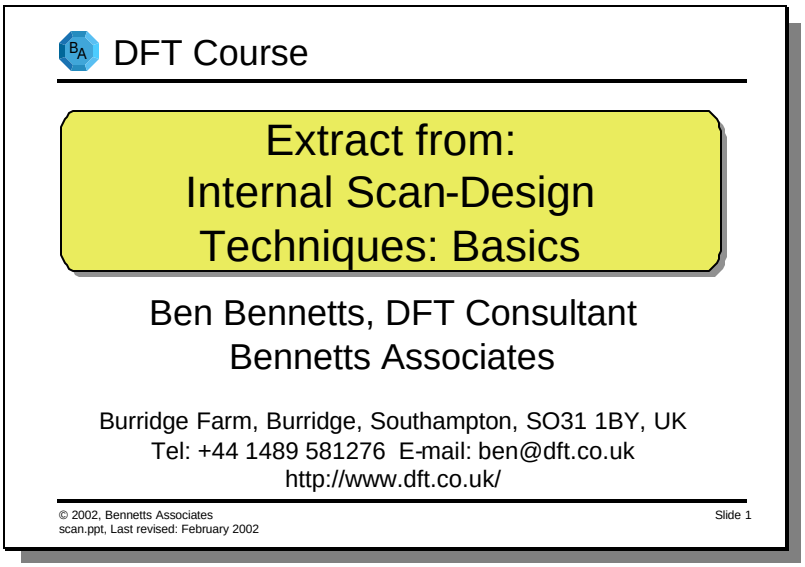
On the Design of Scan-Chains without using Multiplexed D-Type Flip-Flops

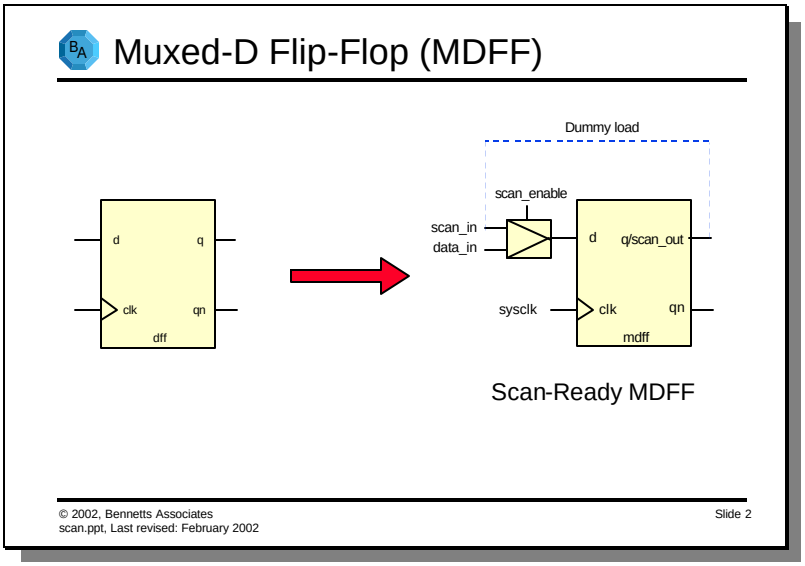
Ben Bennetts, DFT Consultant

22 February, 2002

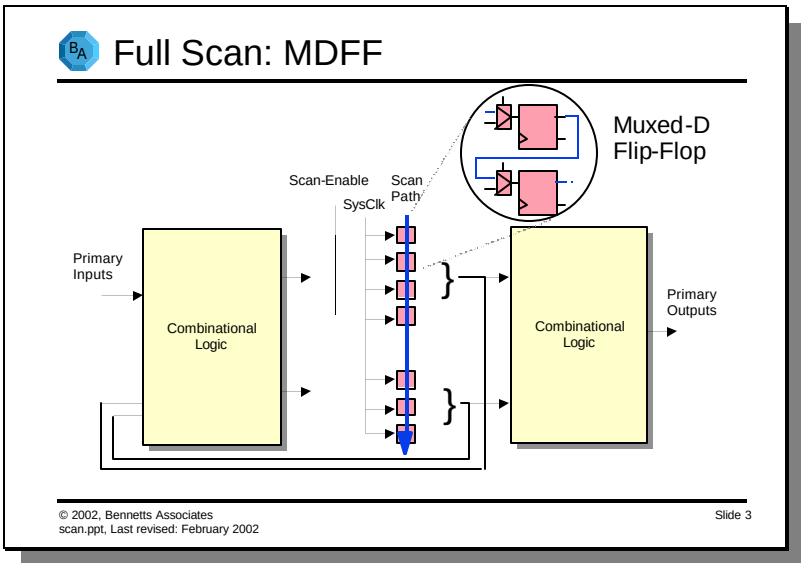


Full scan is a methodology in which the state of all internal state-variable memory elements in a sequential module is made both controllable and observable by an overlaid serial-in/serial-out shift register structure. Effectively, the shift-register structure breaks the feedback path between the memory elements and the combinational next-state and output decoders and allows a new state to be set up independent of the previous state. The insertion of the scan path is done by following a number of scan-based design rules and replacing all the sequential elements in the design by their scanable equivalents. Since all memory elements become observable and controllable, combinational ATPG can be used effectively to generate high fault coverage for the combinational logic blocks — typically close to 100 percent of the stuck-at faults. In addition, full scan has an area and performance impact on the design since scanable elements are slower and/or larger than non-scan sequential elements.

In this extract, we will review the basics of full scan and demonstrate an alternative scan-design technique by working through a scan-design exercise.



The Multiplexed D Flip-Flop (MDFF) approach to scan design is probably the simplest and most popular style of internal scan. Internal DFFs are replaced by a version containing a 2input multiplexer on the data port. One data input is the normal *data_in* port; the other is the *scan_in* port. Data to the *scan_in* port is initially routed from the Q or Qbar output of the same DFF to simulate the extra loading on that output – called a ***scan-ready DFF***. Later, the data to the *scan_in* port will be routed from the Q or Qbar output of the another MDFF to create the shift register path. The routing of data through the multiplexer is under the control of the *scan_enable* control signal.

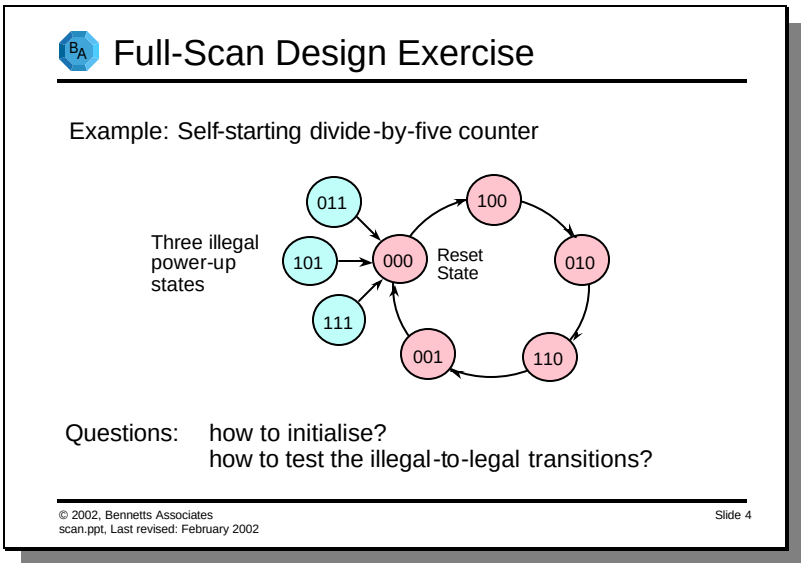


For a full-scan implementation, all DFFs are replaced by their MDFF equivalent and the system clock *sysclk* is used both as the system clock and as the scan clock. *Scan_enable* is routed simultaneously to all multiplexer control lines.

This technique for designing scan chains is supported by a number of commercial scan-synthesis tools and is based on many semiconductor libraries containing MDFF building blocks.

Interestingly, however, this is not the only way to design scan chains into a sequential circuit. A "problem" with the classic MDFF approach is that it automatically locks the multiplexer to the

DFF. We think about the MDFF as a single and inseparable entity with the consequential result that we always see the multiplexer as an additional delay inserted into the mission-mode signals feeding into the FFs. For ultra-high-speed circuits, this could become a major reason not to insert scan chains into the design. The next few slides will look at an alternative approach to scan design that does not incur this penalty.



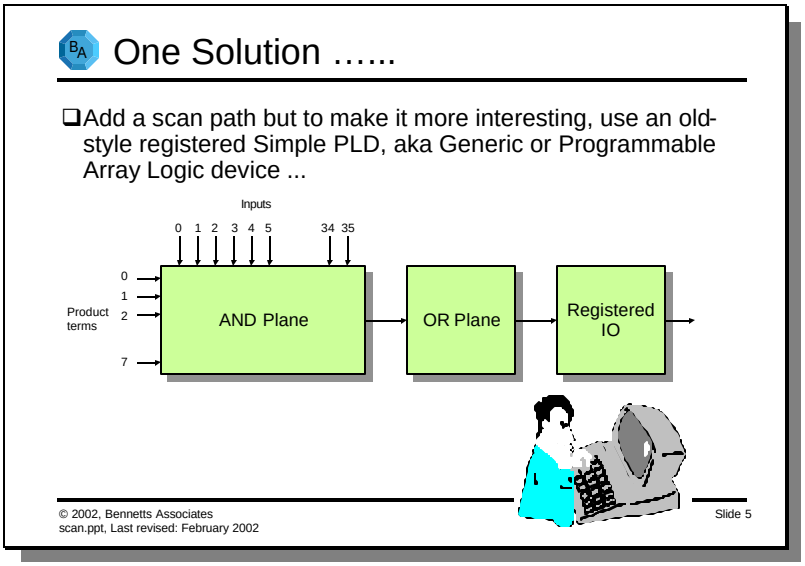
This is a design example taken from a text book on logic design!!

From a test perspective, there are two problems: “How to initialise?” and “How to test the illegal-to-legal transitions?”

On the “How to initialise?” question, this is not difficult to solve: power-up the device and apply a maximum of four clocks whilst monitoring the outputs for entry into the 000 Reset state. If the device is not initialized after four clocks (worst case power-up is state 100), then the device has a defect that prevents initialization – the worst kind of defect!!

On the “How to test the illegal-to-legal transitions?” question, we first should answer another question – “Is it necessary to test these transitions?”. In other words, is there a defect that can only be detected by testing the illegal-to-legal transitions? If there is such a defect (and Murphy says, “Sure there is”!!), then we must test for this defect simply because the design has no master reset and can power up in any one of eight starting states. Consequently, when we test the device as a device, it can pass but when the device is on a board, it can fail.

So, we have a test problem - how to test the illegal-to-legal transitions?

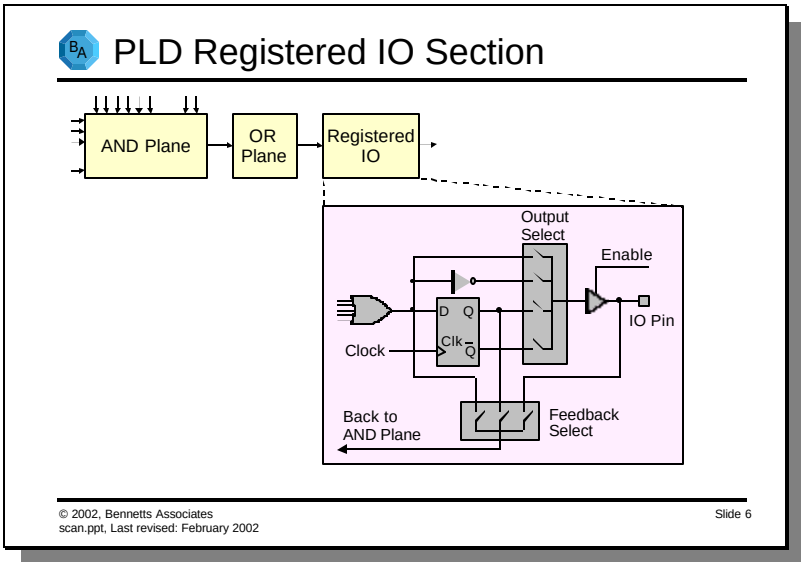


Basically, to solve the problem, we have to make the un-reachable illegal states reachable. We can think of several way to do this butt one solution is to design the circuit with a scan path and then use the path to initialise the circuit into the illegal state followed by a transition to the Reset state. This procedure would be repeated for each of the illegal states.

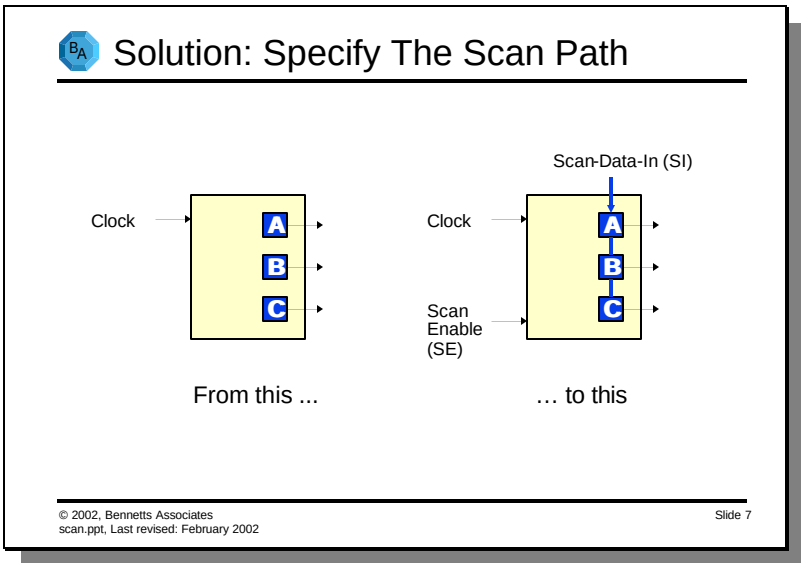
Solving the problem with a scan chain will carry with it all the other advantages of scan-based design and so the class exercise is to design the counter but include the scan chain.

This is a simple design and so to make the design more interesting and to reveal a basic fact about scan design techniques, we will design the self-starting divide-by-five counter using a registered I/O PLD (what used to be called a Simple PLD or PAL device). The question is - how do we design a scan path in a registered I/O PLD? It is not possible to modify the design of the built-in DFFs to include a front-end multiplexer on the *data-in* port. In fact, is it possible to design a scan-path through existing DFFS without adding a front-end multiplexer?

The solution is to modify the original specification of the counter to include a scan-path facility i.e. to synthesize the circuit to include a scan path in one pass through the synthesis process – *1-pass scan synthesis*.

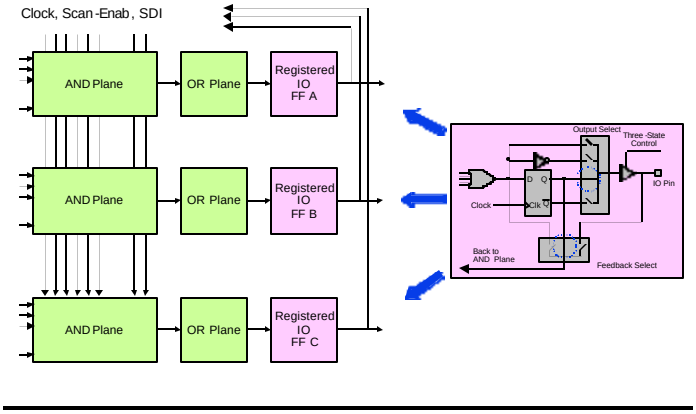


The figure shows the detail of a simple ***Registered IO PLD*** (modeled on an early Altera device), containing a FF with two sets of configurable switches.



The scan-path routing is arbitrarily chosen to be from FFA to FFB to FFC. Two extra inputs are required – *scan-data-in* (si) and *scan-enable* (se).

Need Three MacroCells



scan.ppt, Last revised: February 2002 Slide 8

In terms of the registered IO PLD, the design will need three ***macrocells*** with the output and feedback select switches selected as shown.

Before you move to the next slide, have a think about how to design the circuit. Some words of advice: "You only get what you ask for.", and "There are two ends to a piece of wire."!!

State Table, Including The Scan Requirement

Present State	Future State: Normal se = 0	Future State: Scan se = 1
a b c	a^+ b^+ c^+	**a^+ b^+ c^+**
0 0 0	1 0 0	**si a b**
0 0 1	0 0 0	**si a b**
0 1 0	1 1 0	**si a b**
0 1 1	0 0 0	**si a b**
1 0 0	0 1 0	**si a b**
1 0 1	0 0 0	**si a b**
1 1 0	0 0 1	**si a b**
1 1 1	0 0 0	**si a b**

Principle: synthesise the mux function in the Next-State Decoder

Excitation Equations

$a^+ = \overline{se}\,(\bar{a}\,\bar{b}\,\bar{c} + \bar{a}\,b\,\bar{c}) + se.si$

-> $\overline{se}\,(\bar{a}\,\bar{c}) + se.si$

$b^+ = \overline{se}\,(\bar{a}\,b\,\bar{c} + a\,\bar{b}\,\bar{c}) + se.a$

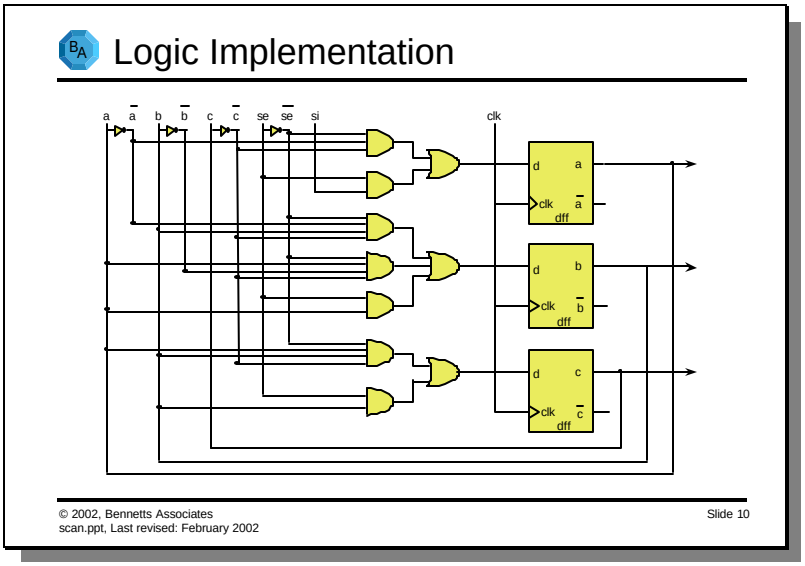
$c^+ = \overline{se}\,(a\,b\,\bar{c}) + se.b$

scan.ppt, Last revised: February 2002 Slide 9

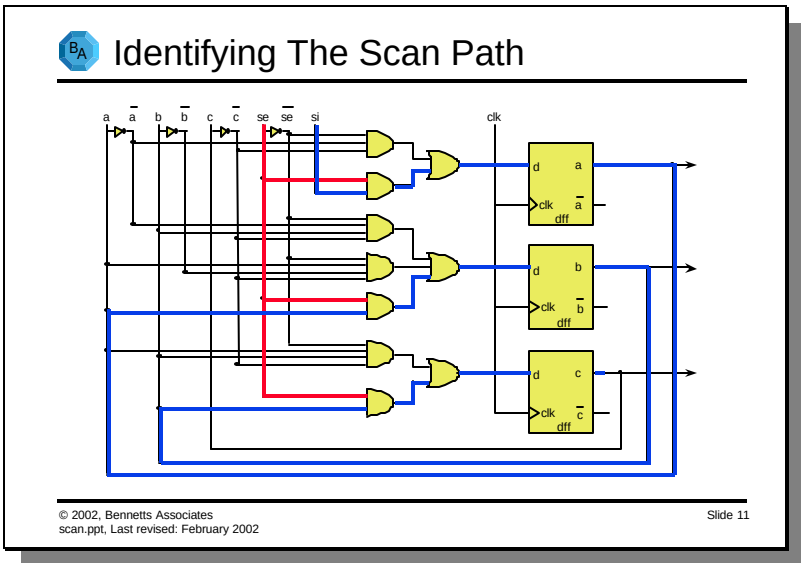
The principle of the design approach is quite simple. First, we recognize that the combinational multiplexer function normally associated with the DFF can be moved back in to the combinational output decoder ("There are two end to a piece of wire."). Second, we have to change the specification of the counter to include the shift-register function as an integral part of the design specification ("You only get what you ask for.").

The modified state table is shown above. SE (Scan-Enable) is arbitrarily set at SE = 0 for normal counter behaviour, and SE = 1 for scan-path behaviour.

Classic state-machine logic synthesis will produce the excitation equations shown in the figure and the logic implementation is shown in the next slide.



Logic circuit for the self-starting divide-by-five counter, modified to include a scan chain and implemented in a registered IO PLD.



The final circuit, shown above, has the multiplexer switching function incorporated into the combinational next-state decoder logic. The scan path is high-lighted. Note that this technique has no impact on performance - the multiplexer is not in line with the signal from the next-state decoder to the FFs - but, once chosen, the routing of the scan path cannot be changed, not unless the circuit is re-designed.

This technique is perfectly general. By approaching the design through a simple PLD, we have been forced to go back to basics rather than automatically assume the MDFF approach, as supported by commercial test-synthesis tools. So, in general, we could design scan-paths into any design simply by specifying the requirement as part of the original design intent at behavioural VHDL or Verilog levels. The synthesis tool will then take care of the synthesis of the circuit in the normal way.

If there is a conflict between FF placement for optimum mission-mode performance and scan-chain routing, the circuit could first be designed down to layout without the requirement for a scan

chain and then, based on feedback from the layout tool that advises on the best routing of the FFs, the circuit is redesigned but now with the scan-chain specified. Layout-tool feedback is usually in the form of a Physical Design Exchange Format (PDEF) file: see Cadence Envisia PKS tool, for example.

--ooOoo--

ABOUT THE AUTHOR

Dr R G "Ben" Bennetts is an independent consultant in Design-For-Test (DFT), consulting in product life-cycle DFT strategies, and delivering on-site and open educational courses in DFT technologies.

Previously, he has worked for LogicVision, Synopsys, GenRad and Cirrus Computers. Between 1986 and 1993, he was a free-lance consultant and lecturer on Design-for-Test (DFT) topics. During this time, he was a member of JTAG, the organization that created the IEEE 1149.1 Boundary-Scan Standard. He is a member of the Board of Directors of ASSET InterTech and the Technical Advisory Board of Teseda.

Ben has published over 90 papers plus three books on test and DFT subjects.

Bennetts Associates, Burridge Farm, Burridge, Southampton, SO31 1BY, UK
Tel: +44 1489 581276 Fax: +44 1489 579519
E-mail: ben@dft.co.uk Web: www.dft.co.uk