Running scan test on three pins: yes we can!

Jocelyn Moreau, Thomas Droniou, Philippe Lebourg, Paul Armagnat
STMicroelectronics, Imaging division
12, rue Jules Horowitz – BP 217. F-38019 Grenoble Cedex, France
Tel: +33 4 76 58 50 00 Fax: +33 4 76 58 67 09
jocelyn.moreau@st.com, thomas.droniou@st.com, philippe.legbourg@st.com, paul.armagnat@st.com

Abstract
Imagers are pretty little objects nowadays, their size is always shrinking and having only three standard digital pins available on their package is a most common thing. Looking back in 2006, only three years ago, people asked for a solution to run industrial structural test on such complex devices could though only reply “impossible” or “Do It Yourself”. STMicroelectronics did not escape the rule. An internal development and a partnered development were thus successively launched to address this issue. This article proposes to examine all the why and how of these developments along with the good results obtained during that time, in terms of test cost improvement, area overhead in silicon, design flow updates and industrialization process. Getting all sensors designed today equipped and test data volume (and time) improvements in the range of 25X to 30X just took that three years time. Now that the solution is industrially available, it’s also time to share and look at the future of industrial scan test on three pins...

1. Introduction
For many years now, the world-wide standard solution for structural digital test has been based on scan methodology. Any standard scan implementation is defining scan chains (with scan-in and scan-out ports), scan control signals (scan-enable, scan-reset) and scan clocks for shifting and capturing. The need for such signal availability is leading to a minimum of one scan-in, one scan-out, one scan-clock, one scan-reset and one scan-enable to be controlled by the Automated Test Pattern Generation (ATPG) tool through test ports, directly accessible from the Automated Test Equipment (ATE). That gives a minimum of five pads or pins to be available on the die or on the package. How to handle scan test with less than five pads, still keeping a correct test coverage and an acceptable data volume on the ATE?

This paper presents the research and developments done in the past three years within the design-for-test (DFT) team of the Imaging division in STMicroelectronics Grenoble (France) to find solutions for scan test on products having as few as three digital pads only. After a review of context and constraints, it will detail two solutions: one based on Synopsys compression architecture and one based on Mentor Graphics’ TestKompress architecture. Both solutions are proven on silicon.

In the rest of this article,
• a functional pad or pin is defined as a pad or pin requested for the functionality of the device on the customer application
• a test pad or test pin is defined as a dedicated pad or pin used for test purpose only (not used on the customer application).

The Imaging division is designing and industrializing CMOS sensors and Image Signal Processors (ISPs). The latter are standard low power digital products, and even though they took advantage of the low-pin-count scan solutions (as explained in the Results section), the driver of this request for such low pin count test solutions was CMOS image sensors, just called “sensors” in the rest of the article. Sensors are low-power mixed-signal products made of mainly a matrix of pixels (analog) and a video processing pipe (digital). This makes them system-on-a-chip (SoC) circuits of mid-range complexity.

The Figure 1 shows a typical sensor architecture with its interfaces.

Figure 1 : Typical sensor architecture
Only the (relatively small) digital part of the sensor (blue boxes) is addressed in the article. Gate numbers can vary from 100 K to 500 Kgates, and translate, in terms of scanned registers, to a range of 8 K to 40 Kflip-flops.
The external signals we find systematically at both die and package level on all sensors are, to date:

- an I2C interface for communication with the application (SCL and SDA signals),
- an external power enable control, switching off or on the whole system.
- an external clock (CLK)
- power supplies and grounds.

Then we find an interface for transferring the image, which can be:

- a serial high-speed link (CCP-2, or CSI-2 protocol, minimum of two pairs of low-voltage differential signals on four pads)
- and/or an 8-bit or 10-bit parallel interface

Defining and developing test solutions for sensors must take into account a few constraints, as explained below.

**Constraint_1:** in many cases, both types of interface (parallel or serial) are available at die level, but in some cases only the serial interface is available, and at package level only one interface is available (serial or parallel).

**C_1:** the scan solution having to cope with any interface configuration, in the worst case we end up with

- I2C interface, clock and serial interface or
- I2C interface, clock and parallel interface

**Constraint_2:** the products also embed hard mixed macros, with scan already inserted and scan chains having a defined length of 100 registers.

**C_2:** the scan solution can not use scan chains shorter than 100 registers.

**Constraint_3:** when the parallel interface is available, scan can be applied on four to five scan chains, using Synopsys Adaptive Scan compression solution. But what about products without a parallel interface?

For such cases, until 2006, a few test pads were also available at both die and package level and were used for test mode programming as well as for scan test, allowing to run scan on two scan chains without compression (not supported at that time by Synopsys with only two scan chains).

However these test pads added on the Input/Output ring and on the package were not well accepted by designers (adding constraints for IO ring on already pad limited designs), nor by customers (adding constraints for application board design). So, the request for finding test solutions using only the functional pads was raised, and gave the next constraint:

**C_3:** no test pad allowed at package level, which means only functional pads can be used to run the tests.

**Constraint_4:** regarding technology, the sensors use specific imaging processes. These are derived from standard CMOS processes but the number of metal layers is limited to four (to minimize the optical stack height). This means that the place-and-route operations can be very quickly cornered by congestion, and the transistors density is quite low (around 50%) compared to standard CMOS (typ. 80%). As a consequence, adding gates for the test purpose can lead to a fast increase of the die size.

The die size is not only limiting the gross margin, but it can also kill the business. Actually, in the camera phone applications, the size of packages is driven by the application, and the die has to fit inside; this leads to constraint_4.

**C_4:** the number of gates added for test solution can not exceed a few Kgates.

**Constraint_5:** the tester strategy of the division is also giving constraints to the DFT; at die level (Electrical Wafer Sort aka. EWS) or package level (final test aka. FT), the memory depth of testers is limited to 10 million vectors (Mvectors) for scan-test (all fault models); this allows 5 Mvectors for stuck-at test, and 5 Mvectors for transition-fault test.

**C_5:** the maximum scan data volume is 5 Mvectors for stuck-at fault model

**Constraint_6:** at final test, the tester is a very low cost platform (home-designed); optimizing its development cost and time means the lesser the number of scan signals to control, the better. In any case the high-speed link interface cannot be used for scan test on this type of tester.

**C_6:** serial high-speed links can not be used for scan-test on the testers used for final-test. It means only I2C interface and the clock are available for scan-test: three functional digital pins only (only taking into account the application pads). This constraint is thus stronger than constraint_1.

**Constraint_7:** on top of these technical constraints, Imaging division is a multi-site design centers organization, and any solution has to be robust in terms of flow, and easy to deploy across sites. Moreover, the design cycle time of sensors is around six months from product definition to tape-out. For these reasons, a single scan strategy has to be defined whatever the products characteristics, in order to be deployed and supported on any site at the minimum cost.

**C_7:** any scan solution has to be defined to be compatible with all the characteristics of the products

These various constraints among products, at both die and package levels, have to be taken into account to define the DFT strategy and make sure the scan solution is applicable in any case. The Figure 2 is illustrating the worst case configuration we have to cope with in terms of pin availability, clearly showing that only three functional
pads (in green) can be used to run digital tests. Power supplies and control (in blue) are not usable, neither the pads for differential link (in magenta).

Figure 2: worst case configuration for pin availability

The scan solution also has to respect the divisional DFT targets and objectives:

- Stuck-at coverage above 99%
- Stuck-at, transition and \( I_{DDQ} \) fault models to be applied in production
- X-tolerant scan compression feature: undefined (X) states entering the compressor do not lead to a compressed X untestable value on output
- Diagnosis capabilities at both die and package level
- As much as possible, same scan patterns at die level and package level to minimize the development time (a single ATPG run for both cases is preferred)
- Minimized test cost
- Compliancy of any DFT solution with the current design flow and minimized impact on development time

We thus end up with the following problem to solve: how to control and run scan test using three signals only (and not test-dedicated), taking into account all other constraints and meeting the DFT targets?

Papers related to compression have been published until very recently [1][2][3] but none is dealing with the specific set of constraints we were facing. The state of the art in 2006 to solve such problems was quite simple: no industrial scan solution was supported by the major ATPG providers (Synopsys, Mentor Graphics, Cadence). Only LBIST (logic BIST) solutions appeared as possibly interesting but presented many limitations as described in [4]:

- required number of gates too big (usually more than a few K gates),
- MISR not X-tolerant (this impacts the time to fix X sources in the design, thus the design cycle time),
- lower test coverage than standard scan solutions requesting test-point insertion and thus adding gates in the design,
- limited diagnosis capabilities, and
- control interface usually based on IEEE 1149.1 protocol, requesting in total more than three pins (JTAG pins are generally supposed to be test dedicated).

Finally the poor level of maturity of such industrial solutions was a risk for not respecting the projects schedule. This solution was thus eventually not even evaluated.

Running standard scan test on three pins appeared as being the only reasonable solution. Three pins for scan means one input (scan-in), one output (scan-out) and one clock. How to manage scan-enable and scan-reset signals? How to manage scan-mode programming? How to reach the coverage and meet the tester memory constraints altogether?

Managing scan-mode programming was actually quite easy without too much effort, as we can see on Figure 3.

The two \( I^2C \) pads are actually used to program test modes thanks to non-scanned registers placed in the test-mode-controller (TMC); once the chip is in scan mode, they become scan-in and scan-out.

Figure 3: Management of test mode programming

But among the five pads requested in the beginning, there were still two to remove: scan-enable and scan-reset.

The following sections will give details on two solutions developed sequentially from 2006 to cope with scan-test on three pins only. The first solution, internally developed, is presented in section 2. It is based on a scan implementation with scan compression from Synopsys (Adaptive Scan). This solution showed advantages and limitations, presented in section 3, that lead to the co-development of a new solution with Mentor Graphics, detailed in section 4. Section 5 will summarize the various results obtained over the three years of development thanks to these low-pin-count scan solutions. A conclusion is given in section 6.

In the following sections, we define three scan-modes:

- Standard scan: scan-test running on \( N \) external channels (\( N \geq 1 \)), with all control signals available from pads, it can use compression or not. It is typically used at die level test if enough pads are available.
- Serial scan: scan-test run on three pads only.
- Bypass scan: scan-test run on a single scan chain gathering all the flip-flops of the design, without compression, requested control signals being made available from pads (scan-enable at least).
2. Internal solution

2.1. Internal design solution

Considering the issues and limitations seen above, ST felt the urge to develop and implement an alternative approach that would fit the needs of the Imaging division.

A concept of serializer/deserializer, highlighted in light red in Figure 4, was imagined (and later published [5] and patented [6], [7]) to overcome the limitation we saw on the number of needed scan signals: a state-machine (Figure 6) was designed to manage the flow of data and route them to the correct destination: either the control signals buffer or the scan inputs and scan output buffer.

Thus all information, either scan data or FSM control bits, get loaded through a single test_in input port at the rhythm of a single free-running tester-controlled clock, sys_clk. The deserializer takes the scan data put in series on one input and gives them in parallel to the original N-channels scan-configured block. Vice-versa, the serializer takes the output of the N original scan-out signals and serializes them to one single output pad. In short, this solution...

- Uses only one pair of pads for scan-in/scan-out
- Allows a “single clock, single scan channel” configuration
- Allows a “zero test-dedicated pad” configuration
- Allows a very high speed scan interface
- Gives the integrator the free choice for his/her test-mode-controller and test-access-point structure
- Allows -as wished- several scan access configurations: parallel, serial, bypass
- Uses a standard synthesis/ DfT insertion/ ATPG flow
- Has a parameterized number of scan chains offered on the Adaptive Scan interface to pads (for EWS multi-channels scan compatibility)
- Builds on simple script to translate patterns from parallel to serial

The diagram on Figure 5 shows the global architecture of what was called an eXtra low pin count Solution for eXtremely compressed Scan (XS^2 in short).

Figure 5: Structure overview for the XS^2 IP

The block diagram above shows the control state-machine, and the shift registers (Shift Data Register and Shift Control Register) used to input the data to the core and create the output flow for scan data from the core.

Figure 6: Test control state-machine driving the XS^2

As can be seen on the Figure 6, the state-machine basically oscillates between SCAN_DATA_SHIFT (where the SDR, the Shift Data Register, is filled in and emptied out), APPLY (where the SDR data get applied/collection, the clock to the core being activated) and, when needed, the
update of the control data in SCR—the Shift Control Register—through the LOAD_SCR state.

The SCR was introduced during the state-machine design to help minimizing the number of pattern cycles that are dedicated to non-scan data (protocol data) transfer. The SCR contains the bits to control scan-enable signal, scan-reset signal, core clock gating, compressor activation and to allow a double clock pulse capture (rpt_vector). Each register, when reset, receives a single ‘1’ (in the most significant bit -Msb) that, after shift, will get detected in the least significant bit (Lsb) as the end flag (scan_end for SDR, ctrl_end for SCR) for the serial loading of that register and consequently as a trigger for machine state change.

To simplify the pattern design flow, it was necessary to have the patterns produced for parallel mode to be reused in the pattern creation flow for serial mode too. This was thus taken into account at the IO ring design: constraints were added in that phase to use IOs with a set of signals gating and conditioning cells in front of them. A special cell was thus built to be used in the IO muxing block, allowing to manage the direction of pads during scan, depending on the chosen test configuration.

In the serial configuration, the control state-machine is thus managing the scan-enable and scan-reset signals, as well as the clock sent to the scanned core for shifting and capturing. It supports the double clock pulse needed in capture for transition-fault test.

Note: since the XS² IP requires that a clock pulse be permanently present for FSM update, it is also compatible with at-speed scan strategies involving PLL activation (not tested today).

2.2. Design flow

As seen, the simplicity and portability of the flow are important. Indeed, XS², seen as an IP, is not a demanding thing. It comes with a documentation including an integration requirements section. The IP is integrated at the top level (Figure 7), aside the core, the IO mux block and the test-mode-controller one. When present, the compressor is also at the same level.

For the remainder, the usual synthesis flow is followed, and the scan insertion is just the same, simply requiring, when parallel interface pads are not available after die packaging, to have a netlist with dummy pads prepared for the ATPG.

![Figure 7: Typical implementation of XS² inside a standard design with Synopsys’ Adaptive Scan compression solution](image)

The front-end design flow summarizes as follows:

- Synthesize the XS² control IP (with Synopsys’ Design Compiler)
- Synthesize the core with its IOmux and TMC.
- Synthesize the top level
- Create the test setup file (for parallel configuration) as SPF (STIL Protocol File)
- Insert the scan and compressor (with Synopsys’ DFT Compiler)
- Save the database

It must be noted here that in order to fulfill the requirement for high-speed scan, only the XS² IP (which is of small area) and the clock tree driving it have to be synthesized for high speed.

2.3. Scan patterns creation flow

The pattern creation flow shown in Figure 8 is as follows:

- Create the standard parallel access test pattern using the netlist with all scan signals accessible
- Save the resulting pattern as WGL or STIL
- Flatten the test pattern (remove the scan option thereof, with a home-made pattern translation tool)
- Post-process the pattern with the dedicated script in order to change the setup phase (one bit to flip vs. the parallel pattern setup from which we start) and transform the parallel loading of scan chains into shift and apply operations for XS² control IP.

![Figure 8: Design flow](image)
The dedicated transcription script is a Perl one and basically processes text files, making it very easy to follow during debug sessions.

2.4. Results

All sensors since the invention of the XS\textsuperscript{2} IP have been implementing it before we could switch to a 3\textsuperscript{rd} party standard solution in late 2008. As such, it proved

- Test-time -wise efficient: test time reduction by 5X when in combination with compressor (compared to a scan run on a single scan chain without compression)
- Easy to implement: less than one week of work to implement it and produce patterns
- Having a fast rising learning speed curve: one month for first implementation to one week after three designs
- Useful for test flow: final electrical test of sensors using scan would have been just impossible without it

A typical implementation uses eight parallel scan chains (to/from Adaptive Scan logic) allowing EWS to occur in parallel configuration, and 48 internal scan chains (to/from the core) with a 5X compression factor. Despite its limitations, described in next section, it was thus a rather good investment in terms of internal design time.

3. Advantages and limitations of XS\textsuperscript{2}

This XS\textsuperscript{2} solution is fully compliant with the constraints and DFT objectives: running scan on three pins only, and fitting in the tester memory. Moreover the solution is using a design flow fully based on Synopsys tools, having thus a minimal impact on the already used design flows. However it presents many limitations:

- It is using an in-house flow, not supported by any EDA vendor
- ATPG patterns have to be serialized thanks to in-house scripts
- Diagnosis is not straight forward (need to convert serial datalog to a parallel datalog format).
- The last point is about the maximum compression we can get with this solution, linked to its architecture using a dual structure (serialization + compression). Is there any risk of being limited in future designs?

We can easily demonstrate that the serial compression value compared to a single scan chain without compression (bypass mode), is equal to the compression we get in a parallel mode when having access to all scan pads.

For such a demonstration, we suppose a symmetrical scan insertion (same number of scan-in and scan-out ports), homogeneous scan chain lengths both in compressed and non-compressed modes; capture cycles are not counted:

\[
C_{\text{parallel}} = \frac{N_{\text{pat}} \ast N_{\text{ff}} / N_{\text{ch}}}{N_{\text{pat comp}} \ast N_{\text{ff}} / N_{\text{sc}}} = \frac{N_{\text{pat}} \ast N_{\text{sc}}}{N_{\text{pat comp}} \ast N_{\text{ch}}}
\]

and

\[
C_{\text{serial}} = \frac{N_{\text{pat}} \ast N_{\text{ff}}}{N_{\text{pat comp}} \ast N_{\text{ch}} \ast L_{\text{sc}}} = \frac{N_{\text{pat}} \ast N_{\text{sc}}}{N_{\text{pat comp}} \ast N_{\text{ch}}}
\]

with:

- \(N_{\text{sc}}\): number of internal scan chains
- \(N_{\text{ch}}\): number of scan channels
- \(C_{\text{parallel}}\): actual data volume compression value in parallel mode
- \(C_{\text{serial}}\): actual data volume compression value on a single channel
- \(N_{\text{pat}}\): number of patterns in parallel mode
- \(N_{\text{pat comp}}\): number of pattern in parallel mode compress
- \(N_{\text{ff}}\): number of scanned flip-flops in the design
- \(L_{\text{sc}}\): length of scan chains in compressed mode

To get the maximum compression ratio compared to a single scan chain, we have to get the maximum compression ratio implemented in parallel mode (whatever the number of \(N_{\text{sc}}\) is). As we noticed on our designs, the compression ratio can be safely increased (with no impact on the test coverage) by increasing the number of channels.

To better assess this possibility of implementing XS\textsuperscript{2} with more scan channels, a practical case was studied and is reported below. We ran an evaluation of this solution on our reference design having 20 Kflip-flops, playing with the final compression value as the parameter.

The Figure 9 shows the volume in serial mode and the scan chain length in compress mode versus the actual compression value in parallel mode.

We also noted above the constraints we have on volume (below 5 Mvectors) and on scan chain length (above 100 registers). We can then extract a working zone, with
an actual compression from 8.5 to 10. To get such values, we need to implement 12 scan channels, thus use 24 pads.

On products without a parallel interface, we can define dummy pads, as seen in section 2, and run all tests in serial mode. However the working area is very small and this strategy did not seem so safe.

On products with a parallel interface, we only have enough pads for eight scan channels. Either we only use XS² (with a non-optimized data volume), or we go for a multimode scan implementation, defining one mode with eight channels to be used in standard scan-mode (at die level) and one mode with 12 channels (using dummy ones) to be used in serial mode (at package level). This solution is demanding two CODEC schemes and it is thus demanding twice the efforts for verification and implementation on tester, and leads to use two different sets of ATPG vectors for die test and final test. So it is not very efficient in terms of development cost.

When looking at these limitations, it was urgent to improve the design flow, and particularly to improve the compression factor of the solution. With the design size increasing, we were sure to meet one day the limit of the XS²-based solution. In one year time, not much progress had been observed on LBIST solutions. We thus got in touch with various EDA providers; Mentor Graphics answered positively for a co-development of a new solution based on TestKompress. The following section, after reminding the objectives and constraints, describes this solution, its advantages and limitations.

4. Mentor Graphics co-developed solution

4.1. Solution overview

The ‘Test Controller Wrapper’ is an industrial solution co-developed by STMicroelectronics and Mentor Graphics [8]. This IP allows a circuit with strong test-pad number limitations to use scan with compression. The compression scheme associated with this interface is the Embedded Deterministic Test (EDT) [9]. Typically, eight signals are required to have an associated pad to use the EDT logic; that doesn’t fit our three pins constraint.

Nevertheless, when associated with the ‘Test Controller Wrapper’, this compression scheme will allow to run compressed ATPG patterns using only three pads instead of eight.

4.2. Motivations

The overall ‘Test Controller Wrapper’ architecture and the results given by this new approach have to satisfy and cover all the constraints defined in section one and overcome all the limitations seen by using XS². The challenge for us was to develop the ‘Test Controller Wrapper’ IP allowing to use only three standard digital pins for scan test on ATE (zero-test-pad strategy) and to be as close as possible to XS² design flow to reduce the impact on design. Using this interface, the data volume of the generated patterns had to reach a minimum compression ratio of 10X at least (as good as XS²) with the same targeted test coverage (99%) as with a standard Mentor Graphics scan approach. The synthesis had to be design-tool independent and the pattern generation flow from ATPG to tester had to remain unchanged and compatible with the tester used. All fault models had to be supported with a same level of diagnosis capability.

4.3. Architecture overview

The Figure 10 gives the implementation of the ‘Test-Controller Wrapper’ IP inside a typical design along with details of its internal architecture, made of five blocks:

- Test controller interface
- Test controller state-machine
- Test controller reconfigurable counter
- Reset mechanism
- Mentor Graphics compression scheme (EDT)

The test controller itself is not scanned and its RTL code is fully synthesizable. Just like the XS² IP, the ‘Test Controller Wrapper’ needs to receive a pulse at every test cycle from the external reference clock. The test controller doesn’t generate any scan clock. It generates signals to gate the test clock by a clock gating cell. The decompressor and the compactor logic (based on one input channel and one output channel) will be added by TestKompress inside the ‘Test Controller’ IP. Around 1K gates is added for implementing the new interface with the EDT logic.

![Figure 10: ‘Test Controller Wrapper’ architecture](image)

* The **interface block** allows the controller to interface with the chip top level.
* The **FSM state-machine block** tracks and defines the different cycles of the ATPG patterns such as load, unload, shift and capture. It also produces all the required test signals (scan controls and EDT controls) for scan test with compression.
* The **counter block** manages the number of shift cycles in the design. A synchronization mechanism is also set in place for managing the transition from I²C mode to scan.
mode to make sure it happens in a deterministic way whatever the conditions (process, temperature, power).

* The EDT is the compression technology from Mentor Graphics for reducing the test data volume and test time, and for increasing the compression ratio with a high level of test coverage. It can work on a single scan channel and is X-tolerant.

* The test-reset mechanism is used for hard-resetting the device at the end of the scan test without powering down the system. This system allows for escaping the scan mode and going back to the I2C normal access mode.

4.4. Configurating modes

The ‘Test Controller Wrapper’ can be configured in three different ways as defined in part one:

- Serial mode
- Standard scan mode
- Bypass mode

⇒ In **serial mode configuration**, the ‘Test Controller Wrapper’ is activated. The signals needed for running scan are generated by the IP through the state-machine. The compression scheme may be activated (or not) since the compression can be used with a one-scan-channel configuration. In this configuration, only three signals require to be driven (one scan-in, one scan-out and one clock).

⇒ In **standard scan-mode configuration**, the test controller is deactivated. The compression can be used with the one-scan-channel configuration. In this configuration, eight signals are required to be driven from pads (one scan-in, one scan-out, one clock, one scan-enable and one scan-reset plus all remaining signals for the EDT).

⇒ In **bypass mode configuration**, the Test Controller Wrapper is always deactivated and relinquishes the control on all test signals. The compression scheme is always deactivated and a reconfigured scan mode (a single long scan chain) is selected. This configuration is mainly a debug configuration, five signals require to be driven (one scan-in, one scan-out, one clock, one scan-enable and one scan-reset).

4.5. Principle of the FSM

The Finite State Machine block embedded in the ‘Test Controller Wrapper’ builds the different cycles of the ATPG patterns (initialization, load, unload, shift and capture). The different states of the FSM are summarized below:

1) IDLE: this is the initial state of the FSM caused by the power-down. All internal signals and the counter are reset
2) SCAN_LOAD_1 and SCAN_LOAD_2: these states define and set the load window and all the signals required for the load procedure.
3) SCAN_SHIFT: this state maps to the shift cycles needed by the EDT decompressor.
4) CONDITION_CYCLE: this is a dead cycle for setting up the conditions for different capture cycles.
5) CAPTURE_CYCLE_(1 to 7): the clock depth (up to 7) is defined here. For patterns where no capture clock pulse is required CAPTURE_CYCLE_SPL is called.

4.6. Design flow description

Each step of the design flow [8] is described hereafter:

- Synthesis and scan insertion: in this step, the design and the ‘Test Controller Wrapper’ will be synthesized with Synopsys’ Design Compiler.
- Generation of the EDT logic: in this step, the EDT logic is generated by TestKompress.
- Synthesis of the EDT logic: Synopsys’ Design Compiler will synthesize the EDT logic RTL to gates and insert it into the netlist. It also writes out a new gate netlist containing the EDT logic.
- ATPG generation: in this step, TestKompress can generate patterns. Two sets of patterns are possible: with EDT on, or with EDT off.

4.7. Enabling the test controller

The power enable signal is used to reset the whole chip including the test controller logic. After reset, the test interface waits for a specific sequence to allow entering the test mode. This sequence defines the test mode programming, the scan shift length, the number of chain test patterns (linked to EDT architecture) and finally the EDT activation control: EDT on or EDT off. The sequence of events is illustrated in the Figure 11.

Figure 11: Specific sequence for enabling the test controller

First of all, a synchronization mechanism is in place to make sure that the modes programmed by I2C are taken into account by the circuit. This is achieved by means of the counter. From the point where the counter is enabled, the ‘Test Controller Wrapper’ waits for 31 cycles. After this 31 cycles delay, it checks the test input. If a low state is detected here, the ‘Test Controller Wrapper’ waits until the test input is asserted high. When this synchronization pattern is found, the interface block records the next 33-bit sequence from the input data pin as configuration data. The first 24 bits define the shift length number, the eight following bits define the number of chain tests and the last bit memorizes the EDT bypass mode use (yes or no). At this point, the scan data are ready to be shifted.
4.8. Results

- Test vehicle

The test vehicle used for the first implementation of the ‘Test Controller Wrapper’ (Figure 12) is a 3.2-MPix device designed in a technology based on CMOS 65nm node. This mixed digital/analog design holds 21 Kflip-flops and the total number of gates is around 500 K.

![Figure 12: Sensor floorplan](image)

- Design results

Regarding the design cost, around 700 gates (0.01% of the digital design area) were added for implementing the new interface with the EDT logic. The core flip-flops have been distributed across 48 internal scan chains.

According to this configuration, the longest scan shift length (scan length + EDT initialization cycles) is 475 cycles. The test coverage achieved at the core level is 99.38% with 2893 patterns. For a scan-based test, the test data volume (in EDT mode) is given by the product of the number of scan patterns times the number of scan cells per chain.

Our data volume is now $2893 \times 475 = 1.4 \text{ Mvectors}$. The data volume in bypass mode is 39 Mvectors.

Compared to a scan run in bypass mode (single scan channel without compression), the compression ratio we get in serial mode with this interface is thus 27.7X, without any loss of test coverage. This value is five times better than the one obtained with the XS² solution.

- Waveforms

![Figure 13: Waveforms of the test signals](image)

The Figure 13 shows the waveforms of the test signals during typical TestKompress pattern generation with the ‘Test Controller Wrapper’.

- Advantages and limitations

The results match all the motivations previously mentioned. The scan is working on silicon in all modes. Data volume and test time have been divided by more than twenty compared to an uncompressed single scan chain implementation. The generated patterns are directly compatible with the tester. No modification of the patterns is required due to the usage of this interface. The main limitation seen today is linked to a design flow not fully automated yet.

5. Results over the three years and outlooks

We summarize in Table 1 the results obtained with both solutions. We ranked the flow easiness from 0 (very hard) to 5 (push-button); flow easiness definition includes design integration, pattern generation and debug on tester.

On top of the limitation on compression results, XS² is also suffering from being an internal solution, without a production flow support similar to what we can get on the low pin count controller (LPCC). It is worth noting that the newest versions from Synopsys (from 2009.06) supports a production serializer; however scan-enable has still to be controlled from a pad, which requests four pads instead of three and thus doesn’t match our constraint C_6. In the same manner, Mentor Graphics is also planning a full automation of the design flow within their test tools.

![Table 1: Summary of the results achieved with the two low pin count solutions](image)

In terms of industrial results, the Figure 14 summarizes the benefits brought by the R&D around low pin count scan.

Both solutions allowed to reduce the pin number for scan down to three pins (violin bars) still meeting the coverage objectives. In parallel the data volume was reduced (red curve). First divided by two with XS² introducing the scan compression (compared to a scan running on two scan channels without compression), it was further divided by five with the introduction of Mentor Graphics’ solution.
Evolution of requested pins number and data volume (normalized) for scan

Figure 14: Benefits of low pin count scan solutions

The development of these solutions, driven by the sensor products were also applied on Image Signal Processor SOCs for specific system-in-package (SIP) products, embedding a sensor and an ISP in the same package. ISPs usually have around 80K flip-flops, a high number of IOs at die level, allowing to run scan on more than ten scan channels. However, once in the package, the number of pins is very limited and we are back to three pins only to test both products inside the SIP. Adding a DFT strategy to control the testability of both die types, the very-low-pin-count solution was embedded in both products, allowing to run scan sequentially on both of them once in the package. The Figure 15 is giving the architecture of such an SIP. The I2C interface, used for scan-in and scan-out once in scan mode is shared between both products.

This approach was fully validated on silicon and is today in production.

Figure 15: SIP architecture

The last but yet untested application we have is related to the implementation of scan-test on in-house very low-cost tester (FPGA-based). The validation of the scan solution is on going at the time of this writing, results to be available by end of Q4’09.

We then foresee a use of such low-pin-count testing solutions for increasing the test parallelism on tester. With only three pins for scan, we drastically reduce the number of requested tester channels to test dice, thus allowing the test of more dice in parallel with the same number of tester channels.

Both solutions are defined to be compatible with transition fault model. Their use with on-chip-clocking (OCC) strategy is being evaluated at design level at the time of this writing, results on silicon expected for Q4’09. Further extensions of this study (e.g. test with two pins) would be possible but do not match our industrial needs.

6. Conclusion

In this paper we presented two industrial solutions to cope with the configuration and execution of scan test using only three non-test-dedicated pins. Both solutions, based on scan compression (Synopsys or Mentor Graphics) are cost effective in terms of design flow (proven to work in a multi-site environment) and test time, with a good test coverage. We could achieve compression ratio from 5X to 25X compared to a scan run on a single channel without any compression scheme. Our internal solution is proven on many industrial designs in production. We highlighted the limitations of the first existing one and presented an improved architecture based on Mentor Graphics compression, also proven on silicon. We demonstrated the added value of such techniques in SIP, and opened many doors for test cost reduction.

References


™ All brands and company names cited here are the sole property of their owner.