Date – 3/30/2010

Attendees: CJ Clark, Bill Tuthill, Ken Parker, Wim Driessen, Adam Ley, Carol Pyron, Ted Eaton, Adam Cron, Dave Dubberke, Roland Latvala, Carl Barnhart, Francisco Russi,

Missing with pre-excuse:

Missing: Bill Eklow, Heiko Ehrenberg,

Agenda:
INIT-DATA – how to achieve persistence

Minutes:

CJ – What we are trying to get down to does it make sense to mandate a particular implementation. And is the persistence even something we can mandate.
Carol – The muxed approach shown was not a mandatory approach
Ted – even holding persistence across TLR can not be mandated. Need board to be able to boot after test.
(slide on 1149.1 view of states in "some persistence after TRST.ppt)
Carol – ken's concern was that INIT process can be time consuming.
Ken was also concerned that something bad could be done to the board if the register wasn't persistent. What he refers to as lobotomized mode.
There are two main modes - System mode and test mode. Power up puts you in system mode. EXTEST puts you in test mode. Bypass would put you back into system mode.
Power Up JTAG is a higher mode than mission mode. Can't say you are in mission mode until you do a TLR assertion. Test Mode can prevent you to get into system mode. TLR cleans it up
Ted – many devices don't have TRST pin and power up in TLR mode without that reset.
Carol – could call that lobotomized mode because you don't know what the chip is doing.
Agree that many chips don't have a TRST pin. It is a requirement for Freescale
CJ – shouldn't focus on TRST pin. It is in functional mode but not clean functional mode.

(slide on I/O Analog Control Setting in "some persistence after TRST.ppt)
Carol – point of this slide. TCK domain and mission mode domain both need to control IO pins. For JTAG to operate you may only need a few permutations. Control with a few registers. In mission mode you have a larger number of settings with more logic.
Ted – is there a difference between interconnect test and mission mode.
Carol – that would be a subset of mission mode
Ted – to operate correctly you may need to set it differently from mission mode
Carol – yes
Carol – Doesn't have to be a mux though. You could have loaded value into function flops with Init Run.

Carol – TRST on flop could be TLR. Not showing persistence.
Carol – init setup would scan register.
In EXTEST or other intrusive instructions follow JTAG rules for IOs
Not in EXTEST follow mission mode rules for IOs

Carol – JTAG flops would persist. They would not be reset.
Agree that the 100 bit case is not burdensome
The case that kept coming up was the FPGA space where you have to scan thousands of bits.
CJ – need to configure the whole FPGA to set the IOs. Once move to INIT DATA register this would be a smaller shift.
Carol – how burdensome is it to redo a setup process?
CJ – would always load the init register even if it was supposed to be persistent just to make sure the right values were being loaded in
Ted – this should only be a recommendation. If your init process time is very large than you should maintain persistence. .
Carol – agree that the functional flops can not be guaranteed persistence during test due to outside stimulus. Not persistence at the pin. Functional logic is changing. As soon as you go back to EXTEST, the JTAG gates are the same and will drive the IOs. Pin is not persistent. The JTAG flops would be persistent and would need to rerun INITRun.
Ted – not test ready after TLR. Functional flops are in reset
Carol – when not in any formal EXTEST mode you are not
Ted- any device that requires init run can not guarantee to be persistent after you enter Test Logic Reset.
Carl – persistence needs to be at the functional level.
Ted – if we could do what you said we could guarantee persistence but it is out of scope.
Changing the behavior of a million flops in my design.

Carol – part of the original discussion was ken's ICT do a TLR at the being of all test sequence. Due to hardware possibly glitching. Traditional way for safe board testing was to start with TLR.
Ken – ICT – typically multiplexed machines. Fewer resources than nails. Need to switch resources. Can't guarantee what the nodes on the boards are seeing are static. The order of the tests are variable. Board is not a complete system. So your view of a system as being able to deal with its own problem is not true. System guarantees may not be there for individual boards. Control loops that are controlling power system could be missing.
Ted – all those things are arguing for doing an INIT SETUP before each test.
ICT is a noisy environment and no control over power
CJ – use JRESET to reset logic.
Carol – the reason didn't like the RESTORE , Can't POR the chip if using a RESTORE instruction.
Carl – only people that would have ti issue JREST are board people. Functional people would not be using init setup and init run
Carol – can achieve that state with a random state

Carol – requirement that was made (not 100% worked out) scan your init register once and don't need to scan it again unless you wanted to make a change. To help Ken's multiple reset
Carol – init data only set by INIT_SETUP.
Carol – assumed init-data is a TCK domain set of flops.

Ted – persistence across TLR is nice to have but not required for initialize architecture
Should be a recommendation
Carl – test software won't be able to figure it out and will always rerun it
Ted and Ken need to figure out if this is really a requirement
Ted – makes it easier and more practical to be used during test but impractical to make it a design constraint.
Carl – if you leave the register alone they will normal persist unless there is something like a power glitch
Ted – will get reset by the reset pin
Carl – doesn't have to be that way.
Carl – experience with the chip design side and doesn't see a problem implementing it either way.
(slide Other INIT Process)
Carol – Rule was a rule on init data that it was not cleared on TLR.
Compromise was to say init data register would not be disturbed by TLR. Not a guarantee that ken gets everything he wants. Maybe need a BSDL construct.
Rarely took any votes and work on consensus. Final rules would go through voting with larger audience. Minutes show that there is still some work here.
CJ – Persistence of INIT DATA – argument for large data . are we that bad off?
Don't like this in between state. Would be for persistence all the time. Don't want persistence only in the INIT DATA register

Ken – other INIT Process. Some process involve deep in the IC. Persistence of "ready to test state" through TLR.
Shutdown PLL or switch them to internal references.
Putting power down subsystems in low power state. No heat sinks
SM and Analog systems into power down mode.
Might want to turn on the 1149.x circuitry to do the testing. (.6 receivers in off state)
Setup IO pins properties
This is where some of the time concerns come from. .not just from shifting in bits.

Ken – lobotomized Modal State
Ted – no difference between Lobotomized Modal State and Test Modal State.
Carol – agrees
Ken – can be a board from a larger system.
Ken – Wanted to create a functional station to say that we are ready for test.
As long as we don't have a power down we are going to stay in the Test Modal State until RESETOR or power down.

Ken – Ready for Test modal state is functional state that has properly configured IOs and internals that are ready for test
Ted – Reconfiguring the internal state of the devices is impractical
Ted – going to TLR is the RESTORE
Ken – so you have a Ready To Test modal state.
Ted – yes. called non test modal state.. Can't define restore setting.
Ken – one last point is embedded test activities. Want to schedule an embedded test during functional test.
CJ – understands ken's point. Sees Ted's side . Agrees with Ken that if not just the init data registers needing to be persistent it is a bit burdensome.
Ted – Limiting the control of complex circuitry to a single TDR is a problematic.
Ken – when you leave tlr to do boundary scan work. There is time before the boundary scan can be done. Need to give the device warning
Ted – making all that persistence over TLR is over stepping our bounds. Would make it a Strong Recommendation.. not to mandate.
Ken – in test mode you are not listening to reset. But in ready for test modal state you are listening to resets.
CJ – can offer in standard.

Ted – how do you document persistence. It is a board configuration issue. Some boards may control it correctly others may not.
CJ – Master reset or private restore. Board level reset at chip happens you need to go to non test modal state. Ken is allowing for that.
Ted - how do you know if there is a master reset and how it is controlled. If you issue a TLR you have no way of knowing if the chip supports that
CJ – you may have a point. Walk with baby steps first. May need more understanding of everything .
Ted – guaranteeing persistence? We can't because we don't know.
Ken – instructions are not mandatory.. we would recommend that if there was an init process, there would be rules to follow. But you don't have to do it.
Carl – optional instructions but if you use them than to mandate how to use them.
Want to create a standard way of initializing device for test.
Carol – only register for persistence was INIT DATA register.
Carl – wouldn't support persistence if only Init DATA. Needs to have INIT RUN as well.. if you have to rerun INIT RUN than it makes the issue more complicated and should be dropped
Ted – I agree.

Continue the discussion on Friday's init meeting

Meeting adjourned: 12:22 EST.

Next Meeting: 4/5/2010 11:00amEST

Action Item by Carl to elaborate on concerns that he has with OO s on power pins and any rules that would need to be added to the standard to address those concerns.

Current Issues listed and who will champion that issue.

1 Observe only. – Ken and Carl
1. Directionality linkage. - CJ
2. Power Pins. - Heiko
3. Pairing power pins with functional I/O - CJ
4. Sample / Capture. – Carol (Freescale) & Roland
5. TRST included in PCB level diagram. – Adam L.
6. Slow to Fall/Rise signaling issue – CJ
7. “No Connect” – Ken and Francisco.
8. Device ID – Still needs work
9. Low-Voltage self observe shorts coverage problem – JJ & Intel
10. Init – Carol & Carl

Action Items:

- CJ will post 1149.1 draft on website with line numbers to make it easier to refer to items in discussion
- Comment #10 CJ will take action to look at possibilities to add to the 1149.1WG website a document which shows which standards are based on 1149.1
- Comment #8 CJ will make changes to draft for observe only
- Comment #7 CJ will get in touch with Doug to get input regarding Comments
- Comment #5 CJ will Add a figure and little text to address TRST use with interconnection of components
- Comment #4 Adam L to add comment about TRST. Update figure 6.8
- Comment #3 Adam L will update language for any proposed change for this section.