Date – 10/05/2010

Attendees: CJ Clark, Bill Tuthill, Ken Parker, Roland Latvala, Dave Dubberke, Heiko Ehrenberg, Craig Stephan, Brian Turmelle, Adam Cron, Wim Driessen, Adam Ley, Francisco Russi, Lee Whetsel,

Missing with pre-excuse: Ted Eaton, Carl Barnhart,

Missing:, Carol Pyron, Neil Jacobson, Bill Eklow,

Agenda:
Wrap up discussion about init-restore/init-clamp

Meeting Called to order at 11:11 am EST

Minutes:
CJ sent out updated figure for Clamp-Hold and Clamp-Release

Ken – have RESET* that goes to boundary cells and IO. Today that signal would only go to update cells that have reset or set capability. Standard says to day that when you pass through TLR you can't predict the state of the PO of the boundary register. We as tool people we are not to count on that data being persistent through the TDR.

CJ – Today the reason that you can't guarantee the values in the register that going through TLR state generates a signal that can be used to change states. But if you mask the Reset signal out of the TAP you can guarantee that the registers don't change.

Ken – There may be something outside of the reset that changes the states of the IO

CJ – we are not creating a mandatory instruction, but one that can be implemented and if a designer adds this instruction than he is aware of these issues and will do it in a way to make stable IO. May still need other capabilities to keep IO from changing. But this is a way to mask reset when through TLR state

CJ - If you are implementing CLAMP HOLD , need to take into account that the boundary registers are not going to change

Ken - Need to look at permission 11.3.h
Ken – Need to make sure this is compatible with compliance enables.
Ken – if you want to use INIT-CLAMP
CJ – don't' want to say that. It is over reaching and unnecessarily
Ken – if you want to pass through the TLR state you will need to keep the compliance enable through that state.

Ken – may need wording and rules to say that if you let go of the compliance enables while in TLR you may compromise the CLAMP state and boundary scan cells may change.

CJ – the Compliance Enables are not going to change if they don't' see the reset signal from the TAP. If you block that signal than the circuit doesn't know you are in the TLR state. And the compliance enable pins will not change.

Adam C –are people going to implement only some instructions with CLAMP behavior. Are we going down this "non standard path".

CJ – Need flexibility. IP designer will be need to be able to block out instructions that need CLAMP behavior and those that don't.

Adam C – using this new functionality- I won't tie this global reset from tap. I will pass it through this gating function and use that signal instead?

CJ – yes that is how you would route the reset signal.
CJ – hoping companies (like Synopsys) would implement this in their standard TAP. That way the user will just be able to use it as they normally would and tie signals from TAP to logic.

Ken – can Roland go back to Carol and check to see if she is OK with some of the information here.

CJ – maybe some confusion between TLR and mission mode. When you enter TLR either Device ID or BYPASS are loaded. Mode signal is changed from test mode to mission mode.
We are not talking about Test Logic Reset bringing you into Mission mode. Need MODE signal to say if you are in Mission mode. And Mode signal is controlled by which instruction is loaded.

Adam C- usage model is that if you use CLAMP HOLD it should be followed up with CLAMP RELEASE

CJ – for sure. Can't have one without the other.

Roland – if you are going to CLAMP your outputs you are helping the chip downstream by providing static inputs. How does that help? What is the usefulness

CJ – usefulness is that you can have communication between chips. If you use ATPG patterns and the IOs are not controlled, you could drive into something. This will allow you to put the boundary cells into a safe state and run functional logic.

Lee – Concern – when you OR the two mode control lines you are increasing the chance that the flip flop gets jitter and flop can be set uncontrollably.
What is the need of blocking the reset line just so that you can go into TLR during test modes.

CJ – that is the main objective. We don't' have control over how often the TAP goes into TLR.

LEE – why cant' you use a RunTestIdle state.
CJ – TLRs are a fact of life. Good way to have a known starting point. Want to hold pins during that TLR.
CJ – don't think JITTER is a hazard in this case. The Clamp-hold signal is what will set the FLOP
Lee- modified the definition of the RESET out of the TAP. One definition is through 5 clock TMSs. Another is going through TLR through the reset pin
CJ – the 5 clocks of TMS high is specifically the one we are trying to block.

Lee – if you use the flip flop approach it will clamp for all instructions.
CJ – if you implement it you can CLAMP before or AFTER the instruction.
Provides a way in the standard TAP and the designer doesn't' need to go the extra effort to provide the capabilities and the test engineer has the ability to use it.

Ken – in 6.1.2 of standard (page 20) definition of TLR State. First sentence is what are potentially changing. Test Logic Reset state – the test logic is disabled so normal operation of the on-chip system logic can continue unhindered.

Ken – Need to modify language in standard to accommodate this new functionality.

Lee – Rule about BYPASS register. Selected by TLR. 7.4.1e When Bypass register is selected. The operation of the test logic shall have no effect on the operation of the on chip system logic.

CJ – CLAMP HOLD isn't affecting system logic.

Ken – when you modify the mode signal this may have an affect on the system logic.

CJ – would like to have CLAMP HOLD override the other rules rather than change all the rules individually.

Lee– should fix the violations in the text.
CJ – rule e) BYPASS instruction isn't changing and is not affecting system logic.
Ken – by changing operation of MODE signal you are affecting the system logic.
Lee – system logic can't operation with MODE high.
CJ- that is the intention of CLAMP HOLD
LEE – id and bypass are known to create normal operation of a device.
Ken – that is a fallacy because there is no normal operation anymore after the test.

CJ – CLAMP HOLD is affecting on-chip system logic.
Lee – you are introducing a second method of controlling the MODE line.
BYPASS should take affect immediately. And its affect is blocked.
CJ – possibly reading into e) more than what is there.

Lee – need to reference that CLAMP HOLD affects the DEVICE ID and BYPASS instructions.

Meeting adjourned: 12:28 EST.

Next Meeting: 10/12/2010 11:00 AM EST

NOTES:
ITC meetings
Public meeting at 10:30-11:30am on 11/2
Power Session at 6-8PM on 11/2
Meeting for 11/2 is canceled

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.