Date – 09/Sep/2011

Minutes of the IEEE-1149.1 Working Group Friday meeting

Attendees:
Brian Turmelle
Carol Pyron
Roger Sowada
John Braden
CJ Clark
Dave Dubberke
Ken Parker
Craig Stephan
Adam Ley
Josh Ferry
Francisco Russi
Roland Latvala
Carl Barnhart
Dharma Konda
Peter Elias

Meeting called to order at 8:35 am MST

Current Draft: P1149 1 Draft 20110820.pdf (_clean.pdf)

Agenda/Overview:
Discussions on ECID
CJ introduced POWERMUX concept

Minutes:

ECID:

Carol opened meeting with ECID continued discussions stating the minimum requirements would be one instruction and a TDR register.

CJ asked Carl what he currently has in the draft.

Carl stated that at this point he had some questions. Specifically it used to be real hard to use fuses to read back a chip id.

Carol – Now using E-fuses and fuse boxes are large, for array repairs and also have ECID

- To read fuse box is a slow process
- Functional POR used to read fuses
- Init Setup does not read the fuse box today. Possibly could change this.

Carl – Asked about having a rule that says ECID should be done after the Init instructions?
Carol – Can happen at any time, just need to ensure the ECID is valid or not. Maybe need to mandate that much in the Std. You have to read the value into a register and shift out the entire string.

CJ – Flash or multi programmable logic can also be used for ECID.

John – If part of Init setup requirements then this would be a good place to do this.

CJ – What are the rules for this? I think we should keep ECID separate from Init.

Carol – Don't want to do every time.

CJ – Allow some amount of time to go by either by a test clk or on chip osc to allow a certain amount of time to go by.

- We have to be restrictive. No external resources.
- We cannot depend on anything except power up to get ECID out.
- It is important to collect failure info and assign to a die.

Carol – Today we do this in manufacturing, but we do this without TCK.

CJ – Any actions we do today in JTAG can be done in system logic too.

Carol – There are other considerations. Separate power supplies to fuse areas.

Carl – What I heard from Ken's answer there are sequential and external resources needed. There are enough different ways to do this on chip

- There could be an Instruction, or TDR or PDL proc (iread and store to a value) But PDL could also do significant sequential work. Would FPGA folks complain?
- Objective is to read the ECID out of a chip and how you get there should be loosely defined.

CJ – External resources allowed?

Carl – No, also define restriction to only use chip resources.

Carol/CJ – Separate power supplies for write and read

Dharma – Power up loads into a register and TCK domain reads out

Carol – We are similar

CJ – We use on chip oscillators, 128 bit ECIDs

CJ – I think companies who want on chip control also use ring osc on chip to control the ECID reads. Quite common in security applications where you don't want external control of clocks

Carol – On ring oscillators, I don't think we can require companies to use that approach.

CJ – What ever mechanism it is, it should be on chip and not require external clocks. (Carol – expect the TAP).

CJ – No matter how you implement it, it should be ready to do its function at init setup.

CJ – If you use system clocks that is today non-compliant. (Init run allows system clocks) per 2001 Std. We have a precedence for this.

Carl – Run bist allows this today.

CJ – Not for setting up IOs.

Carol – It's a mute issue. Any other comments.

John – I think a valid bit is a good idea. LSB or MSB.

Carol – LSB near TDO is best choice.

Carl – One or more valid bits?

Carol – Yes 3 bits perhaps.

CJ – Avoid all ones 111.

Carol – Agreed.

CJ – If not valid, scan again?

John – If not valid it is not ready yet.

Carl – We cannot rely on an iread alone. We need to put into a variable and operate on it.

Ken – Arbitrary length ok? CJ – Yes.

Ken – I'd like all chips ECIDs to be read in parallel.

- Carol – That is unlikely
- Ken – Do them all serially?
- Carl – Synchronize some of them?
- CJ – We'd have to be restrictive if they must all be read on one scan.
- CJ – If more flexible, then must allow different number of iapply's to each chip

- CJ – We could do it, but is that too restrictive? Carl, John, Dharma
- Carl – Need feedback from Cisco
- CJ – This is a one time operation.

Francisco – I have a comment.
A user code was restricted to 32bits today.
So we have added multiple user code registers and read out.
Could we allow variable length user code?

- CJ- User code is just the design value.
- Franscicso – We have 4 or 6 user codes.
- Carol – ECID is unique per part and is not coded in the BSDL
- Francisco – We do a form of this in 32 bit segments.
- Carol – 32 bit chunks
- CJ – Some ECID are not ready immediately.
- CJ – We need:
- Bread crumb bits to define a valid ECID is available
- Then once we have rules defined you can map your 32 bit segments to the new read operation.
- Carol – Francisco you are asking for a side note to allow longer than 32 bits
- Francisco – Yes

Carol – Any other comments?

Dave – I'd like to understand this better, to see if new restrictions apply.

Josh – I cannot tell you how bad external clocks are at in circuit test.

Carol – So Carl do you have enough info?

Carl – Question for the group

- We have several instructions that allow time duration.
- Do we want a duration spec for the ECID too? A new attribute in BSDL, etc. but consistent.
- Carol – That sounds ok
- CJ – John Braden asked for a ready bit.
- Carol – No a valid bit
- CJ – Yup, not ready.
- John – If you have an invalid reading multiple times go to the manual and find out what's wrong.
- Dharma – Is this the same as ready bit today? Check for ready and them go read the ECID.
- Carl – That would require two instructions.
- CJ – Run bist is old and not often used, should we be basing new instructions on this?
- Carl – Init run will also have this.

- Load the instruction and then wait fixed duration or # TCKs or poll the ready bit.
- CJ – Like device id – You provide me a proc to read your ECID
- Carl – For on chip state machines you need a wait time.

CJ – You need to write up the 3 scenarios

- Carl – Which way do we want to go?
- Ken – PDL opens the door for complex ways to get ECID out.
- CJ – Complex for us, but should we allow them to define PDL sequence to read ECID?
- Ken – Do we need limits on this?
- Carl – We want to encourage simple solution to read the sequence.
- CJ – Turning on power to a given domain, or to shut it off, the proc seems necessary.
- Carl – You might need a set of procs to get a chip ready for all of this. IC Reset and so on.
- CJ – How do I satisfy what John said. If I fail am I to wait longer?
- Carol – Use multiple bits?
    - Still processing bit
    - Not done yet
    - Ready
- CJ – Then give me an iproc
- Carol – I'd add the wait state.
- CJ – Then folks will use external clocks
- Carl – ECID wait will not support sys clocks
- CJ – I want a dollar for every person that will use sys clocks and translate to TCK duration.

Francisco – Is this similar to device id register?

- Carol – N o this is not like device id.
- CJ – It is not possible Francisco to get this unique value scanned out, so not as simple as device id which has tie offs.
- Francisco – It doesn't have a capture value in design, but it will have in test. Do a flush test?
- Carol – We model the ECID and test it.
- Carl – You may have access to a fuse box example, if public to share with the group.
- CJ – Can Francisco assume designers are simulating the values today?
- Francisco – Yes.
- Carl / Carol – Part of the test bench.
- CJ- Non-volative memory bits are set to one. Production will program in the value that is unique from then on. Die X/Y coordinates, and so on.
- Francisco – Non test mode, and test mode values?
- Carol – No both are test mode.
- Carl – Access through the TAP.

CJ – Move ahead with 3 ways:

- Optional wait state (2 ways)
- PDL proc (1 way)

Carol – Ken and CJ have different issues.

- CJ – It is very important for tracking a die through board test. We need it to be flexible else we lose support from industry.
- Carol – We have chips in production today, but it would be complex proc to read this information out.
- Carol – Existing Init setup is not completely compliant so may need to rework.

Carl – I'll send something out by Monday.

Carol – 15 minutes now. End or let CJ introduce power mux.

Carl – Carol what about sample relaxation. Can you get it out this weekend?

Carol – Yes I'll do that.

POWERMUX:

CJ presented some material introducing the concept of muxing init_data TDR register based on active power domains. (POWERMUX).

See C.J.'s file power-mux.pdf sent to the reflector for details.

Meeting adjourned: 10:15am MST

Action Items:

-

Next Friday Meeting:

- Next week Friday Sep 16, 2011