

Date – 03/18/2011

Minutes of the IEEE-1149.1 Working Group Friday meeting

Attendees:

Dave Dubberke,
Adam Ley,
John Braden,
Brian Turmelle,
Carol Pyron,
Roland Latvala,
Mike Ricchetti
Heiko Ehrenberg
CJ Clark
Craig Stephen

Meeting called to order at 9:30 am MST

Current Draft: [P1149.1 Draft 20110309 clean.pdf](#)

Agenda: CJ gave an informal review of his PDL work in progress.

Minutes:

CJ – We'll discuss PDL from 1687. There are some known problems but you can see where we diverge from them.

Ken – Will this be Annex C of 1149.1

CJ – Yes exactly right. Should be able to cut and paste and drop into Annex C when it is done

CJ – Starting to put a use model together (Fig C-1). To show how it can be potentially used in the Std.

- PDL's for most IC instances
- PDL not required for any given IC
- PDL's for board itself
- PDL's for IC internal levels such as a memory block

Ken – How are we going to coordinate between 1687 and 1149.1 PDL?

CJ – Working to manage this with 1687 group

Ken – We have a PDL like language, and they have their own version

CJ – We are trying to avoid conflicts between the PDL of each Std. We are trying to manage this. Hold off for now.

CJ – It is not likely there will be a conflict since 1687 has its own registers.

CJ – IC's themselves can have internal levels (memory) which have their own PDL files.

CJ – Levels might be confusing to a Dot1 reader.

Carol – Do you want to allow for levels if a user permission is available to cross between the Stds?

CJ – Open to this

CJ – If a permission is given it could allow TCL usage.

John – Wondering whether 1149.1 is required in a 1687 device? We could have 1687 define level1 and Dot1 define level0 to keep them separate and avoid conflicts.

CJ – 1687 doesn't require 1149.1, but it is allowed. It might not be palatable to 1687 for marketing reasons.

Carol – How about Dot7?

CJ – Dot7 hasn't been discussed.

John – There is no reason we should be defining part of the 1687 PDL.

CJ – I don't follow this. Let's get through the presentation. The 1687 allows for a TAP but does not require it.

CJ – Let's get the PDL language defined and then both camps can iron out the little things.

John – Thanks, I'll be quiet now.

CJ – Table C-1 Parameters:

- iReset – TRST On | TRST Off – currently a difference (1687 doesn't have TRST)
- iWrite – register and pins for 1687, Dot1 register only
- iApply – IR
- iEndState – Was dropped, add back in
- Other new instructions:
- iCall, iTarget, iScan, iClock, iClockOverride
- source – file. Ability to pull in multiple files in TCL today. Add to level 0 PDL

C.6 PDL Structure

C.7 PDL Example

- Start with some simple examples for the reader.
- Agreed by all

Ken – Is PDL case sensitive?

CJ – Yes

Ken - Why 'i' prefix on all instructions?

CJ – 'i' meant internal, but possibly not mandatory now.

C.8.2 Stored Scan Frame:

- iRead and iWrite are queued scan frames
- Setting bits in a stored database, no bits are being applied directly

C.10.1 iPDLLLevel Command

- Must coordinate between the Stds to manage the syntax of this.

Examples:

- iWrite
- iRead
- iScope – Needs more work. Design hierarchy

iRead Command:

- 1687 draft is old and needs some cleanup
- In 1687 case sensitive is needed. For dot1 maybe force all upper case for the dot1 parsers
- Carol – Is that VHDL related?
- Brian – Yes
- CJ – It is nice to use upper and lower case. When you get into mnemonics it might be an issue. Table this and think about it to see what we should allow.
- Ken – VHDL has restrictions on underscore usage. Trailing, or two in a row, etc.
- CJ – Tried to capture that must begin with a letter.
- Ken – We inherited this weird behavior.
- CJ – Does it matter in a quoted string?
- Ken – Need to take offline and review the details.

iScope Command:

- CJ - Works along with iCall command.

Register values when not correct length:

- CJ - Should we call this out as an error. What do others think? A value stored into a frame.

- Ken – I'm not comfortable with the overshift.
- Heiko – Should be able to do this with any register.
- CJ – Should it be an Error or truncated? Putting data into a stored data frame.
- John – Truncating is common.
- Carol – Truncating is nice for hex.
- John – I would lean toward a warning but allow it.
- CJ – For Carol's question there is no concept of overshifting for iRead and iWrite in this context.
- CJ – If I assign a smaller number in a larger field then upper bits should be padded to zeros. Think about it. It should be a mistake, and flagged as an error.
- Ken – Maybe ok to truncate upper zeros. As long as only upper bits if all zeros.
- Carol – I concur.
- CJ – OK if 0x3 is thought of 0011
- Carol – Show both examples
- CJ – More examples will be better. Show both allowed and non-allowed truncation examples
- CJ – iWrite example may be better. Allows padding upper bits with all zeros if your number is shorter than the field.
- Carol – msb lsb definitions for upper and lower bits?
- CJ – For Dot1 msb (closest to TDI), lsb (closest to TDO)

CJ – iRead – If you don't define the expected data, then 1687 retains its old definition from prior execution. You can have binary, hex and X's in the expect fields.

Carol – For polling bist you would expect the same value until done.

CJ – My last iRead, then iApply, and iApply again this is sticky. If you add a TCL wrapper then you could do what Carol wants.

CJ - iApply does the work, IR scans and DR scans to get the register data to where it needs to go.

Ken – What if several instructions each call the boundary register?

CJ – That is an issue we need to work out.

- If we want to target the boundary register and example is:
- iWrite instruction PRELOAD
- iWrite AregFieldfromBScanReg 0x55
- iApply – IR

Carol – Add instruction to the iWrite

CJ – Good idea

- # iWrite instruction PRELOAD
- iWrite AregFieldfromBScanReg 0x55 –PRELOAD

- iApply

CJ – Must check with software guys about this.

Carol – Only allow for boundary register

CJ – May happen in other registers too. Accessing the same register from different instructions is common. This is problematic in an IP subblock.

Carol – Can add –PRELOAD, -SAMPLE –EXTEST

CJ – Let's explore this.

- Discussion 1687 can mix and match writing internal registers.
- Dot1 using the boundary register cannot do this.

Meeting adjourned: 11:00am MST

Action Items:

Next Friday Meeting:

- Next week Friday March 25, 2011