Date – Tuesday July 20th, 2010

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

Missing with pre-excuse: Carol Pyron, Carl Barnhart,

Missing: Bill Eklow,

Agenda:
1) Finalize format for binary/hex values. 0H – 0HABE 0B – 0B01XX
2) New: Library/Use statements for MNEMONICS/REGISTER_FIELDS

Minutes:
Meeting called to order at 11:08am EST
Clean up syntax for BSDL extensions.
Missing Closing Syntax ? missing a closing bracket after Register Field.
Creates a comma separated list.

Ted- use Close Parenthesis at the end of register field.
Only have one register. Either INIT STATUS or INIT DATA
CJ – no current vote on it. Single register to make it easier.
Adam L – suggests using the value pair format. Like what was used in the boundary register attribute. Field name and then end – value would be parenthetical list.
Ken – behavior of "to" and "downto". Does it reorder bits?
CJ – yes it would change.
Ken – does this provide a utility for us?
CJ – value is defined as MSB to LSB.
Ken – does INIT STATUS default to MSB to LSB as well? Do you need to define bits, can you use a default?
CJ – only have to describe the bits that are part of the register. Not legal syntax though. Not something we want to support.
Ken – is the bit mapping in () mandatory?
Adam L – The only purpose of the syntax that Ken proposed is to give it a different name. What is the utility for that?
Ted – to assign it a mnemonic.
CJ – doesn't see the need for this.
Adam C – not a valid shortcut.
Ken – if we have trivially small register do we have to describe the bits for trivially small register.
Adam C – we can use the register name in the PDL
Ken – this is BSDL extensions not PDL.
Ted – still may want a mnemonic associated with it. .but agrees with CJ
CJ – corner case that we haven't allowed before in our tools.

Ken – so far all we have is a collection of examples of BSDL. Need to know what are the mandatory pieces and optional pieces of language in order to write a BNF.
CJ – comfortable with direction we are going in currently. Close to having worked out the pieces we need and once we prove that we can do what we need with the syntax we will formalize it.

Do we want to support Verilog style width?
CJ – no.. To make it simpler we should not support. The register themselves tell the width. The register width will pad it out.
Ted – think the point was more what if I wanted a 1 with 104 zeros. Is there a way to do some concatenation?
Ken – if you had MSB of the register you had a 1 and then 104 zeros, is there a short hand?
Ted – this may be a corner case. And might not need to be addressed
Adam C – field is strictly a binary number and padded with zeros. What about for an output value. Are we padding with zeros.
CJ – padded with x's
Ken – how do we deal with x's
CJ – lets hold on that and focus on how we want to do register width notation.

Adam C – shorthand is pad with zeros on left
Roland – has real world register with 105 bits.
Ted – has registers too that could use the shorthand. Would be easier with concatenation
CJ – can set data to 0. And can index into register and set bit to a '1'
Adam C – can you go through the writes in any order
Ted – there is an order in PDL based on iapply

Roland – discussion about 14000 bit register and needed a shorthand notation for this register and many bits needed to be set
CJ – still many solutions to use to set bits. Index , break down register into smaller bits

CJ – don't have a mnemonic than you can put a value in field. Use PDL style syntax. 0H for hex and 0B for binary. X's don't make sense for iwrites.
CJ – in an iread you would have an X

Ken – iREAD would have hex characters and include X?
CJ – yes. X would represent a 4 bit value. Could not be more granular than that.
Ken – hex + X if you need better granularity you go to binary?
CJ – yes.

Adam discusses the example.
Adam C – 5 bit with 0H1 , what would this represent. Would it be XXXX1?
Ken – when a value is less wide than register how does it get padded?
Ted – hex would have to be a 4 bit value. - 0001
Ken – padded to the left with X's when it is smaller than register.

Register width is 5 bits
Ted – Register = 0H1 would equal X0001 binary
Ted – Register = 0B1 – XXXX1
Ted – Register[0] 0H1 – XXXX1

Mnemonics for internal TDRS
CJ – concerned about the size of space. Mnemonics are large
CJ – can we allow the VHDL "use" statement?
CJ – get more bang for the buck if you are allowed to reuse
Ted – how do you map back to the original register?
CJ – don't really know the lengths of the register.

Ted – if we are going to do something complex we might want use 1687 type environment.
CJ – want to use libraries so that we can reuse registers and definitions rather than reinventing them each time.
Ted – a hieratical BSDL would be useful. Should do it all the way and make the boundary register hieratical and ports hieratical. Might be bigger chunk to bite off than we want to do.
Ken – today we use the "use" for including packages that are specified by standard. So that would be a departure from where we have been
Adam L- not true. Use statements can be used for user content such as BSDL extensions
CJ – can have your own USE statements and ATTRIBUTE and CELL construction.
Adam L – not ready for this yet.
CJ – doesn't feel that it is too complex. Same thing as what we are doing, just renumbering instances
Ted – wants the BSDL to be hieratical. Could be more elegantly described in a 1687 environment and doesn't seem need for overlapping.
CJ – less than verbose than 1687. Not too much overlap.
Ted – agrees with Adam L . Not ready for this now.
CJ – people are asking to do this because they don't want to do the mnemonics over and over.
Roland – would like to see example expanded. Would like to understand the advantage
CJ – will expand example and re-present.

Meeting adjourned: 12:08 EST.

Next Meeting: 7/27/2010 11:00am EST

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.