

Date – 8/24/2010

Attendees: CJ Clark, Bill Tuthill, Adam Ley, Heiko Ehrenberg, Adam Cron, Carl Barnhart, Wim Driessen, Craig Stephan, Neil Jacobson, Brian Turmelle, Francisco Russi, Dave Dubberke, Carol Pyron, Ken Parker,

Missing with pre-excuse:

Missing:, Roland Latvala, Bill Eklow, Ted Eaton,

Agenda:

Review Carl's Hierarchical Fields Up/Down approaches.

Minutes: called to order at 11:03

Review of Hierarchy BNF from Carl

CJ - Bottom up list and Top down list added some complexity

CJ - Expected bit positions as optional statements

Suggested that we look at Carl's examples rather than the BNF to help explain the changes made.

Carl reviews his examples that he sent out in email

Carl has two examples to compare

- Hierarchical fields top-down

- Hierarchical fields bottom-up

CJ - Top level elements are where you would want to concatenate elements.

- Added a different approach to solve problem with the register segment.

- What we are looking for is the ability to concatenate the individual segments.

CJ – The register fields content on the down example is adequate for concatenating register.

CJ – likes the INIT_DATA REGISTER_FIELDS description as is.

Carl – syntax allows you to create it either way.

CJ – BNF that I am looking for is to make individual fields optional and not have 2 different ways of defining REGISTER_FIELDS.

Carl – not sure how to write a BNF such that the length and range are simultaneously optional where you can both or neither.

- Can have unused fields or bits.

- Make it impossible to determine the length

CJ – BNF complicated by not showing range

Carl – if there are not any unspecified bits than this is easier.

Carl – can't get rid of syntax that defines the length of the register.

- Doesn't like mixed syntax.

Ken – how much trouble are you causing for tool vendors?

BNF is half of it. The other half is the data structures that need to be setup on the fly

IEEE 1149.1 Boundary Scan Working Group Minutes

Neil – not sure about the point of having multiple ways

CJ – trying to assemble the over scan chain from the scan chain segments.

Need to be able to assemble smaller registers in order to make full scan chain to keep the bits out of the picture.

CJ – if you change the field in the beginning than the UP method would solve the problem.

Looking to not have the fields on the left hand side of the IS

BNF should have option capability of the length of the registers

Carl- had looked at giving BNF options but don't think that will work.

Range instead of length because he could do any swizzling.

Could fold the changes together, but found that there were too many things that didn't work if made the register range at the beginning of the field optional.

CJ – bits are only swizzle in the sub leveled. Not at the top level

The top levels are putting together predefined elements.

Carl – doesn't agree. Unused bits would need to be included. As well as dummy bits.

CJ – don't need to worry about swizzle bits on dummy and reserved. Just fill in bits.

CJ – would just put length on those.

Carl – bottom up and top down were created so that the software would know if the length of the register is able to be calculated.

CJ – Bottom_UP approach for REGISTER_FIELDS. Top_Down approach for REGISTER_SEGMENTS.

REGISTER_SEGMENTS key word is an alternative method to having 2 versions of REGISTER_FIELDS

Carl – syntax for top down is only at top level

CJ – caution at what top level means. The intent is not to define swizzle bits. The intent is to build up a register from elements. Want to take register segments to put together to make a register.

CJ – feels that more time needs to be spent on the BNF.

Carl- let's try making RANGE IS optional. Probably need semantic checks that if the RANGE IS is not used than the bits defined need to add up to the length of the register.

Carl - Need to predefine register length to check against segment lengths.

CJ – not necessary

CJ – hang up is that we need the length because we don't trust the segment definitions.

Need to trust the tools that write the segments

CJ – might want to add the segments together

Ken – is a 1 or 2 level description good enough?

CJ – Time frame for Carl wasn't enough time to get the syntax and examples gelled. We are making progress however.

IEEE 1149.1 Boundary Scan Working Group Minutes

Ken – do I have a good way of expressing a sparse amount of data in a large register?

Multiple definitions of the same bits?

CJ – RANG IS syntax will support these examples.

Ken – 2 BNF's?

CJ – no, making the RANGE IS optional.

Ken – is it optional on all lines of the REGISTER_FIELDS definition?

So when the first line is written a precedent is set.

CJ – doesn't make sense to mix the two types.

Ken – Sees a BNF switch based on the first line.

CJ – the other approach would be adding attribute *REGISTER_SEGMENT* without ranges. And REGISTER_FIELDS would have RANGE_IS

Carl – what I used as the switch was the IS after the register name {init_data *IS* vs. init_data }

Ken – BNF would cover both options by using the *IS* Switch

Ken – BNF was fully recursive. Concerned the allowance of that is unnecessarily complex.

Carl- only the hierarchy is recursive. Only thing that is recursive is the segment and the segment can be used to define another segment.

Ken – would like to see an example of the recursive structure

Carl – it is an n-level hierarchy rather than 2 levels.

Ken – would like to see a reason why we did it.

CJ - Suggests Carl to show another level of hierarchy. Break the IO into quadrants as an example which then call out the XYZ_IO.

Carol – Agrees. Would work with Carl to show more than one level of hierarchy.

CJ – is ok with using a different Attribute if it would make it easier and the group agreed.

Carl will email the example out to the group after the meeting

*Wim is out for the next 3 meetings as he is out on Holiday.

Meeting adjourned: 12:12 EST.

Next Meeting: 8/31/2010 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.