High Frequency Response Tracking System micro-architecture

Gopalakrishnan Sridhar, Senior Logic Design Engineer/Micro-architect, Intel Technology India Pvt Ltd, Bengaluru, India (gopalakrishnan.sridhar@intel.com)

Vadlamuri Venkata Sateesh, Logic Design Engineer, Intel Technology India Pvt Ltd, Bengaluru, India (vadlamuri.v.sateesh@intel.com)
Agenda

1. Introduction
2. Response Tracking System
3. Complexities in meeting Higher Frequencies
   A. push a request entry into the RTS
   B. pop from the RTS during response
4. Summary
5. References
Introduction

Why Response Tracking System is Required?

- To track the completions of the Requests initiated by the Bus master
- To frame the proper completion format.
- To check for the unexpected completions
- To track the remaining byte count.
Introduction (contd..)

What are the complexities with Response Tracking System?

- Dynamic buffer management is required, as the static queue allocation for each tag is area prone.
- Response of different tags can come out of order.
- For every new transaction going out of bus master, has to create a link with the existing linked list.
- As response of each tag comes in the same order, so RTS should be capable of removing that entry from the linked list.
- Has to provide the its attributes on a “tag”.
Response Tracking System

1. During Bus Master Request
   - Set the tag valid, head/tail pointer and link list
   - Store command attributes

2. During Response to the bus master
   - Provide the transaction attributes for the given tag
   - Remove the transaction form the cpl tracker on rsp_pop assertion

© Accellera Systems Initiative
### Response Tracking System Functionality

#### Tag and Valid

<table>
<thead>
<tr>
<th>Tag</th>
<th>Valid</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
</tr>
</tbody>
</table>

#### Head

- Tag: 2
- Valid: 0
- Next ptr: 3
- Head: 9
- Tail: 9
- Next ptr: 7
- Valid: 13

#### Tail

- Tag: 2
- Valid: 0
- Next ptr: 3
- Head: 9
- Tail: 9
- Next ptr: 171
- Valid: 13

#### Current ptr and Next ptr

<table>
<thead>
<tr>
<th>Current ptr</th>
<th>Next ptr</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>7</td>
<td>11</td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
</tr>
</tbody>
</table>

#### Free ptr

- Current ptr: 2
- Next ptr: 3
- Next ptr: 5
- Next ptr: 3
- Next ptr: 5
- Next ptr: 3
- Next ptr: 5
- Next ptr: 6
- Next ptr: 3
- Next ptr: 3
Complexities for meeting Higher Frequencies in Response Tracking System

During a entry push into the RTS:
- RTS has to check whether its buffers are full or not and then update tail pointer and split counter.

On arrival of response, evicting an entry from RTS:
- RTS has to find the head of the linked list of a given tag
- And then find the split count and attributes of the head pointer.
- Based on split count and attributes decide to remove the entry or not.
Entry push into RTS

- Pushing the entry after buffer full indication check from RTS leads to huge combinational path.
- Pipelining command push will lead to improper functionalities.
Micro-arch solution For Entry push

• Instead of using buffer full information from the RTS, check with one outstanding transaction counter, before pushing the transaction.
• It will not create functional implications, though command input interface is pipelined.
Complexity during Response pop from RTS

• Popping out entry from the RTS after checking the split count and other attributes checks leads to huge combinational path.
• Pipelining on the rsp_pop directly will lead to improper functionalities.
Micro-arch solution for Response pop from RTS

• Pipe line the all attributes separately.
• It has to pop the entry for the previous response, and to get the attributes for the current response of the same tag.
Summary

Response Tracking system with the micro-architectural solutions provided in this paper can be deployed in High Frequency SoC/NoC Designs
References


Questions?

Thank You