AMBA Port aggregators

[1 AMBA Master Port Aggregator 3](#_Toc430706160)

[1.1 Introduction 3](#_Toc430706161)

[1.2 Specifications 3](#_Toc430706162)

| Revision # | Author | Date | Notes |
| --- | --- | --- | --- |
| 0.1 | Joji Philip | 22 Sep 2015 | Initial Draft |
| 0.2 | Joji Philip | 24 Sep 2015 | Add details on the handling of width conversion. Port reflection. |
|  |  |  |  |

# AMBA Master Port Aggregator

## Introduction

Currently in an AMBA NoC, each AMBA host master port requires a dedicated master bridge and dedicated router ports for attachment to the network. Master bridges can be area intensive when considering large AID tables, reorder buffers, clock crossing and multi-VC switches. Master port aggregator allows several master ports to be aggregated into a single master port which then connects to the NoC through a common master bridge. This allows the master ports to share logic for packetization, clock conversion, ordering, switching etc. This also allows a host with multiple master ports to connect to the NoC through a master bridge at a single grid point instead of spreading out over multiple grid points with a master bridge per port.



## Specifications

### Port types

Currently only AXI3, AXI4 and AXI4-Lite source master ports can be aggregated. Aggregated master port is of AXI4 type.

A source master port of AHB or other type can be considered by instantiating a converter on the source port output and connecting the converted AXI4 master port to the aggregator.

### Port Properties

#### Data widths

Each source master port can have a different AXI data widths. Aggregated master port’s data width is the maximum of the source master port widths. However no AMBA width conversion is performed in the port aggregator.

#### Address widths

Each source master port can have different address width. Aggregated ports address width is the largest address width of the source master ports. Smaller address widths are padded with 0 for higher order bits.

#### Clock property

Port aggregator can have clock domains for each source ports and one for the aggregated port. Each source port can be specified and synchronous, asynchronous or ratio-synchronous with respect to the aggregated port’s clock domain. This is just provided for flexibility, the general expected use case for low area would be to have all source ports and aggregator port to be in a single clock domain so that no clock crossing modules are required. Master bridge attached to the aggregator can always be used to perform clock domain crossing.

### Quality of service

One important aspect that needs attention in the design of port aggregator is fairness and isolation among the source master ports and maximize utilization of the aggregated port bandwidth. There is arbitration among the source master ports to send on the aggregated master port. There is an arbiter for AR channel and one for AW channel.

Sections 1.2.3.2 to 1.2.3.6 are low priority features.

#### Round robin

This is the simplest form where each source master port has equal priority to access the aggregated port. AR and AW arbiters select among the source master port in a work conserving round robin order.

#### Weighted round robin

If source ports need weighted share of the aggregated port, then a fair WRR arbitration can be used where the weight of source ports is configurable through NoC Studio.

#### Port Rate limiters

Rate limiters can be instantiated on AR and AW channels of each source master port to limit the amount of traffic a given master can inject into the NoC. The rate limits can be configured through NoC Studio. Rate limiter module currently deployed on the Tx switch interfaces can be reused here.

#### Write store and forward

Once a source port is selected on W channel, it holds the aggregated port’s W channel till it sends out WLAST. A slow rate port can affect the throughput maintained on the aggregated port and hence unfairly affect other higher rate source ports waiting for the aggregated port. An option is to enable store and forward on a source master W channel. If SAF is enabled on a source master port, its AW channel participates in arbitration only if the W channel has received WLAST or a configured amount of WDATA. In SYNC or ASYNC mode, this instantiates a buffer on W channel which performs store and forwards functionality. VC buffers used on the routers have this functionality built in and should be reused. Note that store and forward buffers on the W channel make more sense if splitting to a burst size is implemented as well.

#### Read response buffer

Each source master port’s read response channel can optionally instantiate a read response buffer which allows it to absorb read response bursts from the NoC. This prevents the aggregated read response port from being blocked by a slow source master exerting back pressure on its read response channel.

#### Optional transaction splitting

Long transactions from source master ports can again cause fairness issue among the ports. An option is to split long transactions into multiple shorter transactions at a configurable address size boundary. This functionality is currently built into AR and AW channels of the master bridge and can be reused.

#### Max outstanding limit per port

Each source master port has configurable maximum number of AR and AW transactions that can be outstanding in the NoC. These are implemented as counters incremented when a transaction is accepted and decremented when corresponding response is completed. While a port has saturated its max outstanding limit, it doesn’t participate in arbitration.

### Functional design

#### Block Diagram



#### AID calculation

Each source master port can have a different AID width. These are normalized into a container equal to the maximum source AID width by padding msbits with 0 for narrower AIDs. Each master port also has a designated port ID. AID sent on the aggregated port is a concatenation of the two.

{Port ID, Normalized source AID}.

Hence the AID width of the aggregated master port and hence the attached master bridge is equal to 4-bit port ID + Max source AID width.

#### Port width conversion

As mentioned before, aggregated port’s data width is the maximum of the source port data widths. Transactions from narrow sources are sent as AXI narrow transfers on the aggregated port.

##### Handling data widths for write data

Write requests from narrow agents are sent as AMBA narrow write requests on the aggregated port. Write data from narrow agents needs to be steered to appropriate output byte lanes based on write transaction start address. Each subsequent input beat is steered to shifting output byte lanes to generate AXI narrow write transaction. Note that the granularity of de-muxing the source data is equal to the width of the source agent.

For example, there are three source agents with AXI data width 32-bit, 64-bit and 128-bit. The aggregated port is 128-bits wide in this case. When the 32-bit agent is selected, its data is steered to 1-of-4 32-bit segments on the 128-bit output interface based on the start address, every subsequent input beat is shifted to the next 32-bit segment. For 64-bit source, data is steered similarly to 1-of-2 64-bit segments of the output data. 128-bit source’s data goes as is on the output interface.



##### Handling data width for read response

Read requests from narrow agents are sent out as AMBA narrow read requests on the aggregated port. Read response from the master bridge will return in narrow read response format on the aggregated port’s read response channel. Appropriate data segments from this wide data must be provided to the destination narrow agent’s read response port. However the segment to be selected depends on the original read requests start address. To avoid storing this information in the port aggregator, the master bridge is required to 0 out all invalid data byte lanes on narrow read responses. With this, narrow output data can simply be generated by folding the data segments and ORing them. For e.g. read response to a 32-bit source port would be bit wise ORing of the 4 32-bit segments in the 128-bit aggregated port read data channel. For a 64-bit source agent, read response data would be ORing of the 2 64-bit segments in the 128-bit aggregated port read data channel. For a 128-bit source agent, there is no folding and ORing, its read data channel is same as the incoming read response channel of the aggregated port.



### Master bridge configuration

### AMBA link pipelines

Each source master port can specify a configurable number of pipeline stages. Pipeline modules are added on the AXI channels and both READY and VALID paths are registered.

### Output register

A NoCstudio controlled parameters decide addition of optional register stage at the command channel outputs on aggregated port and read/write response outputs to

### Port replication

This functionality checks that traffic on two source master ports are exact replica of each other every cycle. Two ports marked for checking will have exactly same configuration and pin outs. AR, AW and W have to be exactly equal every cycle, any mismatch must be marked as an error status. Traffic from only one of the ports is actually sent out to the aggregated port. Response traffic from the aggregated port is mirrored on both source ports.



### Low power

### Register bus access