Address Space Decode New Functionality Proposal for Ncore 3.2/3.4

Table of Contents

Table of Tables

Introduction

Agent interfaces for Ncore 3 implement address space decoding, transactions will be directed to a target for completion based on the aperture within the global address space.
Each aperture is described by a set of 3 General Purpose Region Definition Registers, two registers are required to define the base address of a region (GPRBLR = AddrLow, GPRBHRx = AddrHigh). The base address of any aperture is aligned to 4096 bytes.

Table 1: Aperture Base Registers

Bit	Name	Description	Access	Reset
31:0	AddrLow	Low order bits 43:12 of a region's base address	R/W	0x00000000

Bit	Name	Description	Access	Reset
7:0	AddrHigh	High order bits 51:44 of a region's base address	R/W	0x00
31:8	Rsvd	Reserved	RO	0x000000

The third register, GPRARx, defines the region attributes. The number of apertures is configurable by the user, a special (default) boot region exists and uses the same register format. A similar set of registers exists in DCE.

Table 2: Aperture Attribute Register

Bit	Name	Description	Access	Reset
0	Rsvd	Reserved This bit had been defined as the ***Hazard*** flag in the PCIe document and was no longer required	RO	0x0
1	ReadID (ARID)	No ordering by AxID (free listing) for reads[1]	RW	0x0
2	WriteID (AWID)	No ordering by AxID (free listing) for writes[15]	RW	0x0
4:3	Policy[1:0] 11 Endpoint Order 10 Relaxed Order 0X Reserved	Policy determines the setting of the internal OR[1:0] field which defines the behavior of the agent at the bottom of Ncore Endpoint[2] order: ▪ All transactions to the same peripheral region are issued and completed in order ▪ Override the ordering mode of the incoming transaction to *Endpoint Order* Relaxed order[3]: ▪ Responses within a flow (same AxID) are ***ordered*** if ReadID/WriteID bits are zero, otherwise they are ***unordered*** ▪ ***Ordered*** transactions will be issued as soon as the previous transaction in the ordered sequence has been ordered at the target (StrReq has been returned) and command credits are available ▪ ***Unordered*** transactions will be issued as soon as command credits are available at the target Write: BRESP may be returned (earliest) as soon as DtwRsp has been received, following AXI ordering rules (all older responses have been sent)	RW	0x0
5	NC	Non-Coherent 0: Use coherent mode as indicated by incoming transaction • CHI/ACE/ACE-Lite • For AXI interface default transaction mode depends on a configuration parameter 1: Enforce non-coherent transaction • For AXI interface configuration this setting becomes a NOP if the default configuration is non-coherent	RW	0

[1] Setting ReadID and WriteID (freelisting) will force the internal OR[1:0] field to 2b00 (unordered) for memory targets
[2] Read: Issue --> DtrReq, Response --> DtrReq --- Write: Issue --> DtwRsp, Response --> DtwRsp. Setting ReadID or WriteID bits is not recommended for this policy and may result in undefined behavior
[3] The equivalent of ***Strict Response*** ordering can be achieved by using ***Relaxed Order*** and setting the ReadID and WriteID bits

Bit	Name	Description	Access	Reset
6:5	NSX[1:0]	This field describes the Security Level required to access this region. NSX[0] is mapped to NS(non-secure) NSX[1] is mapped to NSE (non-secure extension, reserved for later use)	RW	0x0
8:7	Rsvd	Reserved	RO	0x0
13:9	HUI	Home Unit Identifier used to access this memory regions - this field uses the index of the target HU within the enumerated set of DMI or DII	RW	0x0
19:14	Rsvd	Reserved	RO	0x0
24:20	Size	This field describes the size of the aperture in address space. The size of a region is calculated as: size_of(IG) * $2^{(Size + 12)}$ with IG = Interleave group	RW	
28:25	Rsvd	Reserved	RO	0x0
29	Rsvd	Reserved - this bit has been reserved to be used as an extension to HUT when hierarchical gateways or chip-to-chip interfaces will be implemented	RO	0x0
30	HUT	This field describes the Home Unit Type 0: System Memory 1: I/O (Peripheral Memory/Devices)	RW	0x0
31	Valid	This bit indicates if an aperture descriptor is valid 0: Invalid Mapping - this descriptor is not describing a valid region 1: Valid region descriptor	RW	0x0

New Functionality

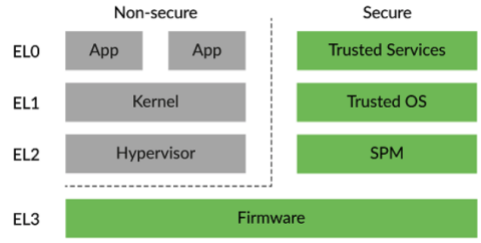
ReadID/WriteID/Policy:	These fields are not applicable to CHI or ACE transactions and shall be reserved in CAIU
NC - bit:	Each aperture shall be configurable to either use of coherent or non-coherent transactions Coherency does not apply to HUI target = DII and the value of the bit shall be ignored for this setting
NSX[1:0] field:	Each aperture may have a security property encoded in NSX[0] = NS NSX[1] has been reserved for later use in Arm v9.0 to support REALM and ROOT security state
NS = 0:	Only secure transactions (with the NS bit on the agent interface set to zero) shall be accepted. All non-secure transactions shall be terminated with an error response, the illegal access shall be reported by the error handling system within Ncore
NS = 1:	All transactions will be accepted by the AIU

Background: New Security states

TrustZone was introduced in Armv6 and provides the following two Security states:

- Secure state
- Non-secure state

The following diagram shows the two Security states in AArch64 with the software components that are typically found in each Security state:



The architecture isolates software running in Secure state from software running in Non-secure state. This isolation enables a software architecture in which trusted code runs in Secure state and is protected from code in Non-secure state.

RME builds on the Arm TrustZone technology. RME extends this model, and provides the following four Security states:

- Secure state
- Non-secure state
- Realm state
- Root state

The following diagram shows the Security states in an RME-enabled PE, and how these Security states map to Exception levels: