BASIC SYSTEM MMU

Functional specification

Document Version	1.0
Release Date	April 20th, 2016

Table of Contents

1 Preamble

System MMU has been developed according to the following specification that provides basic features for a system MMU.

It most likely not usable like that and most probably that IP has to be fine tuned, adapted to the SOC overall memory management, clock management, power management strategies and security policy.

To support Linux, an additional attribute (Access Flag) shall be added in the table.

As well performances can be improved by adding a pre-fetch unit, support of outstanding requests or use of set-associative cache structure for TLB.

2 Overview

2.1 Introduction

System MMU provides 2 main functions :

- Virtual to physical address translation
- Memory protection

System MMU upstream port can be directly connected to 1 master and used as a dedicated MMU or connected to multiple masters via an interconnect and so used as a shared MMU.

It allows non-CPU master to see contiguous memory space and remove the need to allocate fixed physical memory space or to manage memory fragmentation via a scatter-gather DMA.

Memory translation, attributes and validity information are contained in a set of translation tables located in downstream memory.

System MMU supports secure and non-secure context but doesn't support virtualization.

2.2 Feature List

The following features are supported:

- Supports only stage 1 translation.
- Supports secure and non-secure context.
    - Supports only Secure Device issuing Secure transaction to System MMU and non-Secure device issuing non-secure transaction
    - Security State is obtained from security signaling of upstream bus and not from

address space.

- Supports Stream matching
    - Memory translation and memory access operations are done only for upstream transactions matching the AxID and Secure information saved in one of SMR registers.
    - In case of unknown stream ID a configuration fault is generated.
- Each TLB way is dedicated to one context (based on stream matching).
    - It means that there is a set of context registers per TLB way.
- Supports only 4KB translation granule. (64KB is not supported)
- Supports 3-level page tables structure allowing to support 1GB, 2MB and 4KB page size.
    - Supports up-to 39-bit virtual address space
- Kernel / User space determination is done thanks to virtual address MSB.
    - if all 1s, Kernel space is selected
    - if all 0s, User space is selected
- Information used for Tagging TLB: Virtual address, ASID (Address Space Identifier assigned by the OS), page size.
- Supports the following table attributes:
    - Secure / Not Secure
    - Access Permission
    - SHarable attributes
- Other attributes like Privilege Execute Never, Unprivileged Execute Never, Access Flag are not supported.

3 Functional Description

System MMU is a fully synchronous design sequenced with a single clock, it's configurable through the AXI4-lite programming interface.

Low power interface is implemented as a Q-channel device that is intended to be connected to a Q-channel controller. System MMU never requests clock or denies request from controller to transition to low power mode.

At reset, system MMU is in bypass mode, meaning that transactions from upstream port are directly

propagated to downstream port.

When system MMU is set in active mode, transaction request (read / write address channel) from upstream port is processed as below (under control of TLB state machine):

1. AxID is compared with all SMR (Stream Matching Register) registers filtered with Secure information. In case of non-matching, transaction is not propagated to downstream port and a “DECERR” response is sent on the response channel. In addition an interrupt is generated indicating a configuration fault.
2. In case of stream matching, the set of context registers and TLB memories corresponding to the stream ID are selected. Upstream address is compared with virtual address stored in tag memory (set corresponding to stream ID) , with ASID0 or ASID1 register is compared with ASID saved in tag memory, and with kernel / user space info with the one stored in tag memory.

a) In case of matching Physical address and memory attributes are retrieved from PA memory. Transaction is propagated to downstream port.

b) In case of non-matching , Page Table Walk (PTW) state machine is activated.

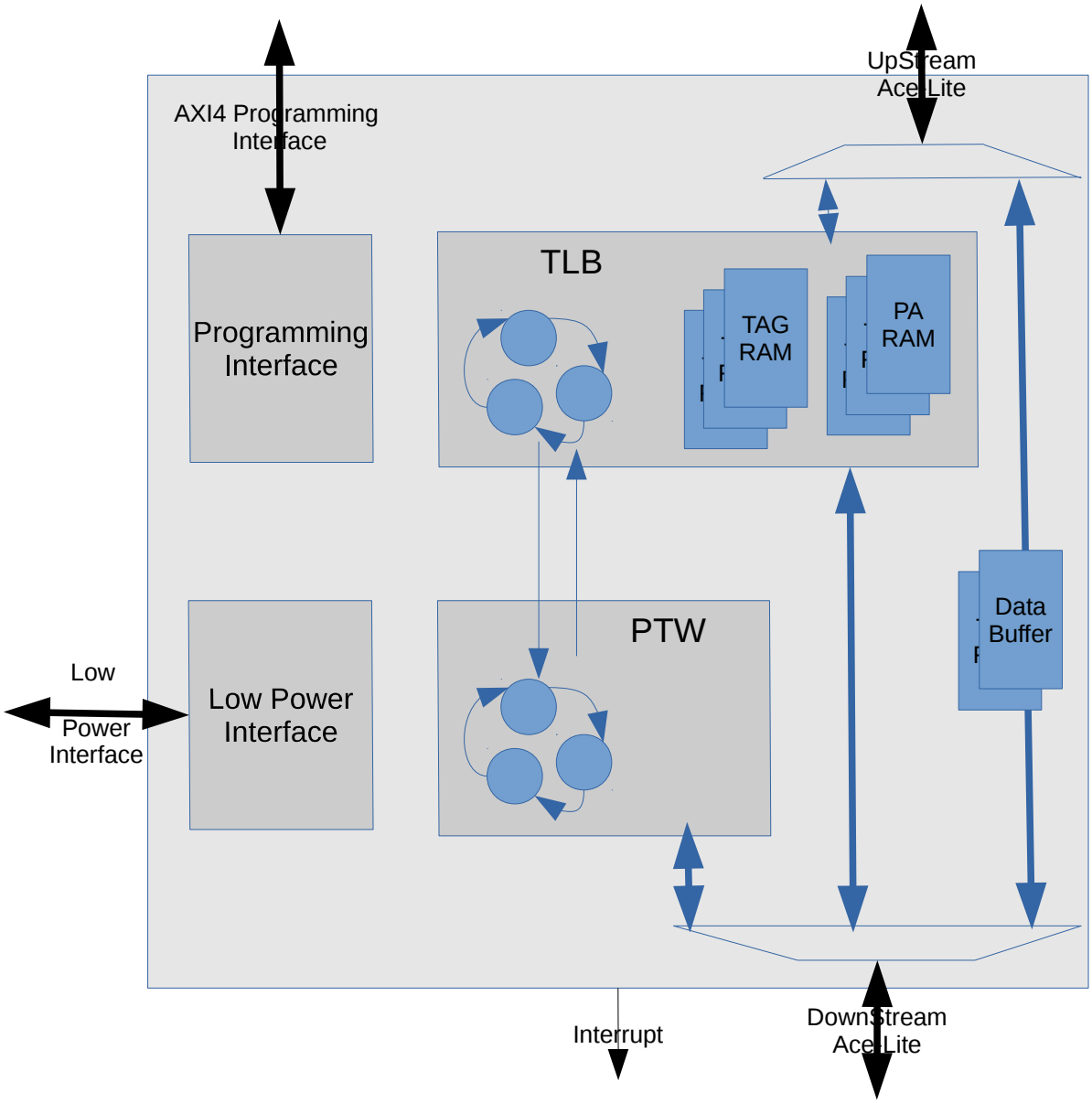
3. Kernel / User address space is determined according to virtual address space (upstream) and TCR_EL1.T1SZ, TCR_EL1.T0SZ values. In case virtual address space falls in unmapped region, a fault is generated.
4. PTW accesses the 3-level page tables through the downstream port.

a) Tables attributes are checked and in case of mismatch between tables attributes and upstream transaction attributes a fault is generated.

b) In the other case, Physical address and attributes retrieved from the highest level table are stored in the the PA memory and the virtual address (upstream), current ASID and page size are stored in the TAG memory. Transaction is propagated to downstream port

Read and Write data channels are buffered between upstream and downstream ports.

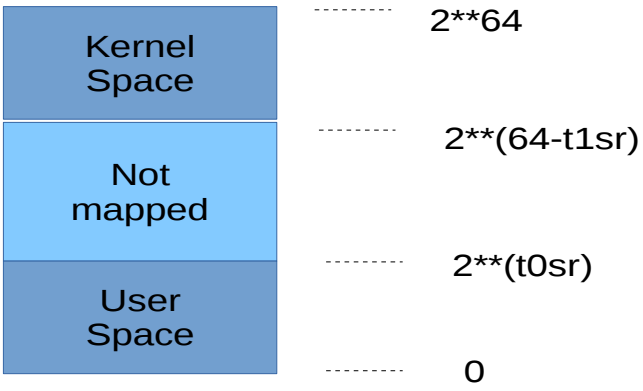
3.1 Top level block diagram



3.2 Virtual address space

Supports up to 39-bit virtual address space.

Below figures show the definition of kernel and user space for a 64-bit bus address.



Virtual address space and tables index mapping is given by the below table:

VA bit[38:30]	VA bit [29:21]	VA bit [20:12]	VA bit [11:0]
L1 Table Index	L2 Table Index	L3 Table Index	Page offset address

3.3 Page table information

Below tables are given for a 64-address bus.

L1 table address is defined as below:

PA bit [63:44]	PA bit [43:12]	PA bit [11:3]	PA bit [2:0]
0s	TTBR1 (Kernel) or TTBR0 (user)	VA bit [38:30]	000

Tables format is defined as below:

63:48	47:16	15:10[1]	9:8	7:6	5	4:3	1:0
R.	Base address of the block or to the next table	R.	SH	AP	S / NS	R.	TT

TT : Table descriptor Type

- 11: provide pointer to the next table address.
- 10: provide block address
- 0X: invalid entry

1 Access Flag is not implemented whereas it's mandatory for supporting Linux.

Version: 1.0

S/NS: Secure or Not Secure

- 1 => Secure / 0 => Not Secure

AP: Access Permission

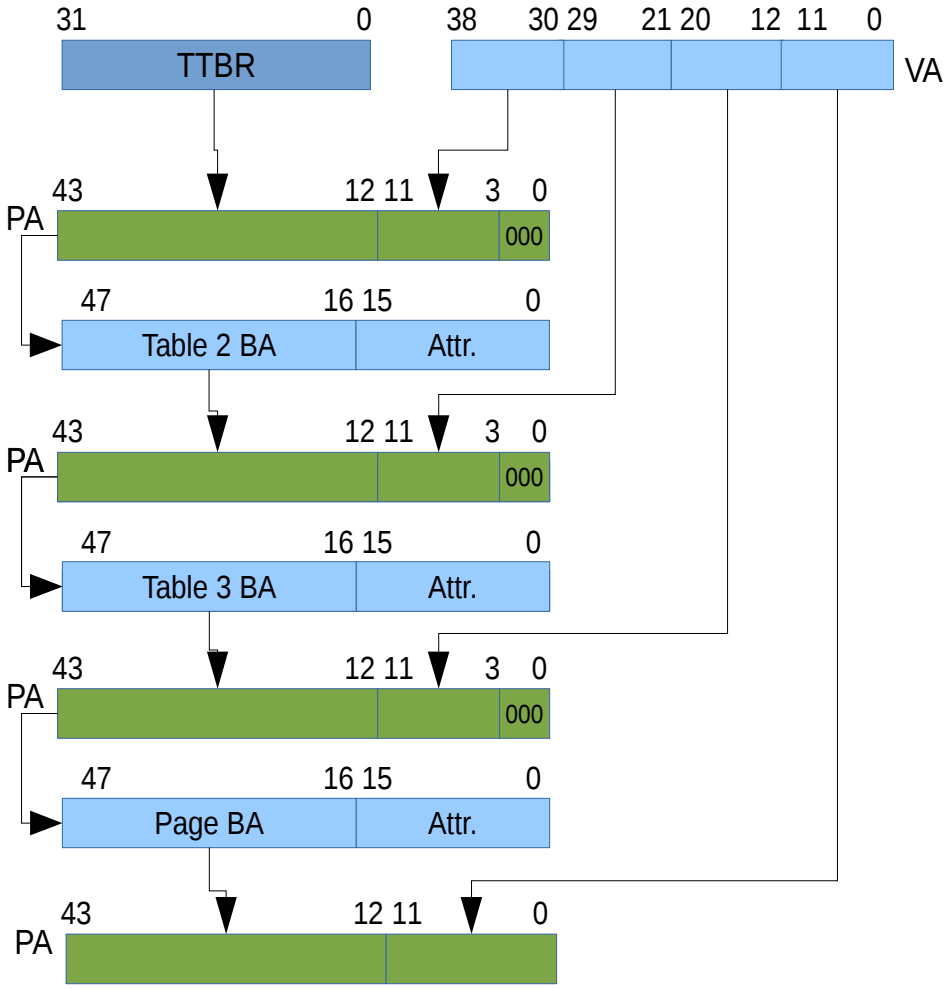
AP	EL0	EL1/2/3
00	No Access	Read and write
01	Read and write	Read and write
10	No Access	Read-only
11	Read-only	Read-only

SH: Sharable attribute

- 00: Non-Sharable
- 01: Inner Sharable
- 10: Outer Sharable
- 11: System

R.: Reserved

3.4 3-level tables translation example.



3.5 Generic parameters

Parameter	Default value	Definition
BUS WIDTH		
BUS_ADD_WIDTH	64	Upstream and downstream address bus width
BUS_DATA_WIDTH	64	Upstream and downstream data bus width
USER_WIDTH	4	Upstream and downstream user info bus width
BUSC_ADD_WIDTH	12	Programming address bus width
BUSC_DATA_WIDTH	32	Programming data bus width
ID_WIDTH	4	Upstream and downstream AxID width

Parameter	Default value	Definition
MMU Parameter		
STLB_WAY_NBRE	1	Number of secure TLB way
TLB_WAY_NBRE	2	Number of non-secure TLB way
TLB_ADD_WIDTH	8	TLB memories depth = 2**TLB_ADD_WIDTH
ENC_MIN_PAGE_WIDTH	12	Granule page size = 2**ENC_MIN_PAGE_WIDTH
VA_WIDTH	39	Virtual address width
L1_TABLE_ADD_WIDTH	9	L1 Table index with
L2_TABLE_ADD_WIDTH	9	L2 Table index with
L3_TABLE_ADD_WIDTH	9	L3 Table index with
ASID_WIDTH	16	ASID register width
KERNEL_USER_SEL_WIDTH	16	Define the number of bit used for the Kernel / user space selection
FIFO_DEPTH_ENC	4	Read and Write data channel buffers size: 2**FIFO_DEPTH_ENC. If downstream and upstream data throughput differ, buffers size shall be set to the max burst length supported by the read and write channels.

4 Register Description

4.1 TCR_EL1

There is one TCR_EL1 register per TLB way or per context.

Address base for all TCR_EL1 register: 192 (decimal)

Address for TCR_EL1 with context n is: 192 + n*4.

Accessible only in privilege mode (kernel)

	31:30	29:28	27:23	22	21:16	15:14	13:12	11:6	5:0
	R.	SH1	R.	A1	T1SZ	R.	SH0	R.	T0SZ
		R/W		R/W	R/W		R/W		R/W
Default		0s		0	0s		0s		0s

R.: Reserved
T0SZ: Size of the User space = 2**T0SZ
SH0: Sharable attribute of downstream port for PTW access using TTBR0 (User space)
T1SZ: Size of the Kernel space = 2**T1SZ
SH1: Sharable attribute of downstream port for PTW access using TTBR1 (Kernel space)

A1: For Kernel space, select ASID1 when equal to 1 and ASID0 when equal to 0.

4.2 TCR2_EL1

There is one TCR2_EL1 register.

Address for TCR2_EL1 is: 0

Accessible only in privilege mode (kernel)

	31:17	16	15:9	8:5	4:1	0
	R.	TLB_TAG_INV	R.	PTW_AxID	PTW_QOS	BYPASS
		WO		R/W	R/W	R/W
Default		U		0s	0s	1

R.: Reserved
U. Undefined
WO: Write Once – Reading returns always 0
BYPASS: MMU is bypassed when set to 1.
PTW_QOS: QOS attribute of downstream port for PTW access.
PTW_AxID: Read Access ID of downstream port for PTW access.
TLB_TAG_INV Start the invalidation process of TLB TAB entries when set to 1.

4.3 FAR (Fault Address Register)

There is one FAR register.

Address for FAR is: 4

Accessible only in privilege mode (kernel)

	31:27	26:0
	R.	FA
		RO
Default		U.

R.: Reserved
U. Undefined
FA : Fault Address.
In case of MMU fault, the address of the access having caused the fault is stored in that register.
Page address offset is not stored
FA[26:0] = VA[38:12]

4.4 FAR2 (Fault Attribute Register)

There is one FAR2 register.

Address for FAR is: 8

Accessible only in privilege mode (kernel).

	31:10	9:6	5:3	2	1:0
	R.	FAxID	FAxPROT	FRnW	FS
		R0R	R0R	R0R	R0R
Default		0s	0s	0	0s

R.: Reserved
U. Undefined
R0R: Reset On Read
FS : Fault Status

- 00: No fault
- 01: Configuration fault
- 10: Table fault
- 11: Reserved

FRnW: Indicates if the access having the fault is Read access (1) or Write access (0).
FAxPROT: Store the AxPROT attribute having caused the fault.
FAxID: Store the AxID attribute having caused the fault.

4.5 SR (Status Register)

There is one SR register.

Address for SR is: 16

Read only register.

Accessible only in privilege mode (kernel)

	31:27	0
	R.	TLB_TAG_INV
		RO
Default		0

R.: Reserved
RO. Read Only
TLB_TAG_INV Invalidation process of TLB TAB entries is in progress when set 1.

4.6 TTBR0 (Translation Table Base address Register - el0)

There is one TTBR0 register per TLB way or per context.

Address base for all TTBR0 register: 32 (decimal)

Address for TTBR0 with context n is: 32 + n*4.

Accessible only in privilege mode (kernel)

	31:0
	TTBR0
	R/W
Default	0

R.: Reserved
TTBR0: provide a pointer to the first level table for user space. Used by PTW.

4.7 ASID0 (Address Space IDentifier - el0)

There is one ASID0 register per TLB way or per context.

Address base for all ASID0 register: 64 (decimal)

Address for ASID0 with context n is: 64 + n*4.

Accessible only in privilege mode (kernel)

	31:16	15:0
	R.	ASID0
		R/W
Default		0

R.: Reserved
ASID0: Address space identifier for user space access (set by the kernel)

4.8 TTBR1 (Translation Table Base address Register - el1)

There is one TTBR1 register per TLB way or per context.

Address base for all TTBR1 register: 96 (decimal)

Address for TTBR1 with context n is: 96 + n*4.

Accessible only in privilege mode (kernel)

	31:0
	TTBR1
	R/W
Default	0

R.: Reserved
TTBR1: provide a pointer to the first level table for kernel space. Used by PTW.

4.9 ASID1 (Address Space IDentifier - el1)

There is one ASID1 register per TLB way or per context.

Address base for all ASID1 register: 128 (decimal)

Address for ASID0 with context n is: 128 + n*4.

Accessible only in privilege mode (kernel)

	31:16	**15:0**
	R.	ASID1
		R/W
Default		0

R.: Reserved
ASID1: Address space identifier for kernel space access (set by the kernel)

4.10 SMR (Stream Matching Register)

There is one SMR register per TLB way or per context.

Address base for all SMR register: 160 (decimal)

Address for SMR with context n is: 160 + n*4.

Accessible only in privilege mode (kernel)

	31:4	**3:0**
	R.	SM
		R/W
Default		0

R.: Reserved
SM: Stream Matching. MMU Context is selected when upstream AxID access match SM value. If none of the context SM values match the upstream AxID, an fault error is generated.