APB SLAVE SRAM CORE

PRODUCT DESIGN SPECIFICATION

Version *1.0*

17 June 2018

Author: Farshad

VERSION HISTORY

This project was developed for educational purpose by Farshad. GIT version control software was used for monitoring and tracking the project development phases.

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	*Farshad*	*May 20, 2018*	*Farshad*	*June 17, 2018*	Initial Design Definition draft

UP Template Version: 12/31/07

Table of Contents

1 INTRODUCTION

1.1 ARM AMBA 3 APB V.1 PROTOCOL

Advanced Peripheral Bus (APB) is a bus protocol developed by ARM to provide low cost interconnect protocol with low power consumption and minimum complexity. Details regarding AMBA APB protocol can be found in ARM_AMBA3_APB.pdf or in ARM's official website.

1.2 ABOUT AMBA APB SRAM CORE

This project is licensed under MIT open source license v3.0.

The APB Slave SRAM core is a slave peripheral core of AMBA APB Master Bus. It is a Static Random Access Memory core for storing data through write operation and retrieving data through read operation via APB Bus interface. This Ram core also supports error response and wait state, which can be useful for integrating this core with different design requirements.

2 GENERAL OVERVIEW AND DESIGN GUIDELINES/APPROACH

This section describes the principles and strategies to be used as guidelines when designing and implementing the system.

2.1 ASSUMPTIONS / CONSTRAINTS / STANDARDS

The aim of this project was to develop an APB complaint memory core, and reusable APB Master Verification IP. This project assumes that user will configure the SRAM core and the testbench to fit their project needs.
The following assumptions were made while developing the SRAM memory core:

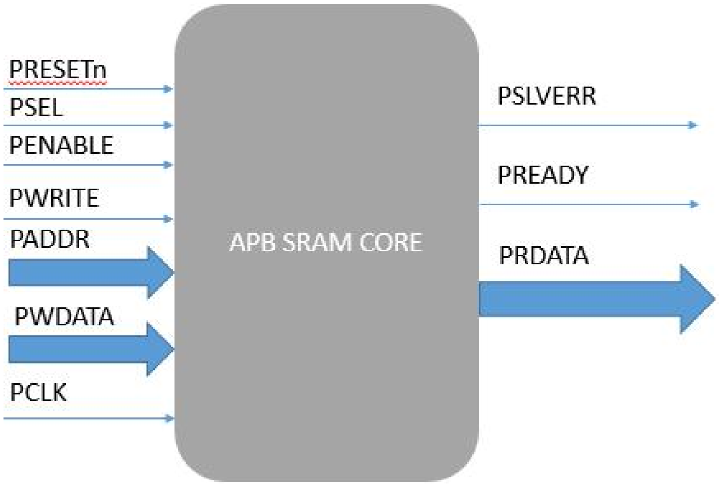
1. The memory block has to be resizable. That means you can define how much memory will be assigned per memory address such as, 8 bit, 16 bit, 32 bit etc.
2. The memory size must be configurable. That means you can define the address space of the RAM like 0 to 63.
3. The Address and data bus width must be configurable. You can define the bus width for address and data bus.
4. The design must implement configurable wait state that can be enabled or disabled based on user demand.
5. The slave must trigger error response whenever memory read or write operation is out of bound,

3 ARCHITECTURE DESIGN

This section outlines the system and hardware architecture design of the system that is being built.

The SRAM core interface is developed using AMBA v3 APB v1 specification which can be found in ARM_AMBA3_APB.pdf

3.1 BLOCK DIAGRAM



3.2 HARDWARE ARCHITECTURE

3.2.1 Configurable Parameters

3.2.1.1 Memory block and memory size

The SRAM core memory block and memory size can be configured via MEMSIZE and MEM_BLOCK_SIZE parameter respectively. The default value is set to MEMSIZE = 64 and MEM_BLOCK_SIZE=8.

3.2.1.2 Address and Data bus width configuration

Address and data bus width can be configured using ADDR_BUS_WIDTH and DATA_BUS_WIDTH parameters. The default value of these parameters are set to 32.

3.2.1.3 Reset value configuration

The reset value of the memory address can be configured via RESET_VAL parameter. By default the value is set to 0.

3.2.1.4 Wait State configuration

The wait state functionality can be enabled by setting EN_WAIT_DELAY_FUNC parameter to 1. By default it is set to 0.

When Enabled the slave core will generate random wait state delay between the read and write operation transfers which can be controlled by MIN_RAND_WAIT_CYC and MAX_RAND_WAIT_CYC parameters. By default these parameters are set to 0 and 1 respectively.

3.2.2 IO ports

Port name	Width	Direction	Description
PRESETn		I/P	Active low reset signal
PCLK		I/P	Clock signal of 100MHz
PSEL		I/P	Slave Select Signal
PENABLE		I/P	Enable signal
PWRITE		I/P	Write Strobe. 1 = Write, 0 = Read
PADDR	[ADDR_BUS_WIDTH-1:0]	I/P	Address Bus
PWDATA	[DATA_BUS_WIDTH-1:0]	I/P	Write Data Bus
PRDATA	[DATA_BUS_WIDTH-1:0]	O/P	Read Data Bus
PREADY		O/P	Slave Ready Signal
PSLVERR		O/P	Slave Error Response Signal

3.2.3 Reset Operation

Reset operation can be performed by setting PRESETn signal low for at least 1 clock cycle. Upon reset all memory should hold the reset value specified by RESET_VAL parameter.

3.2.4 Write Operation

Write operation is performed according to the procedure defined in APB specification.

The following transfer diagrams are taken from ARM_AMBA3_APB.pdf

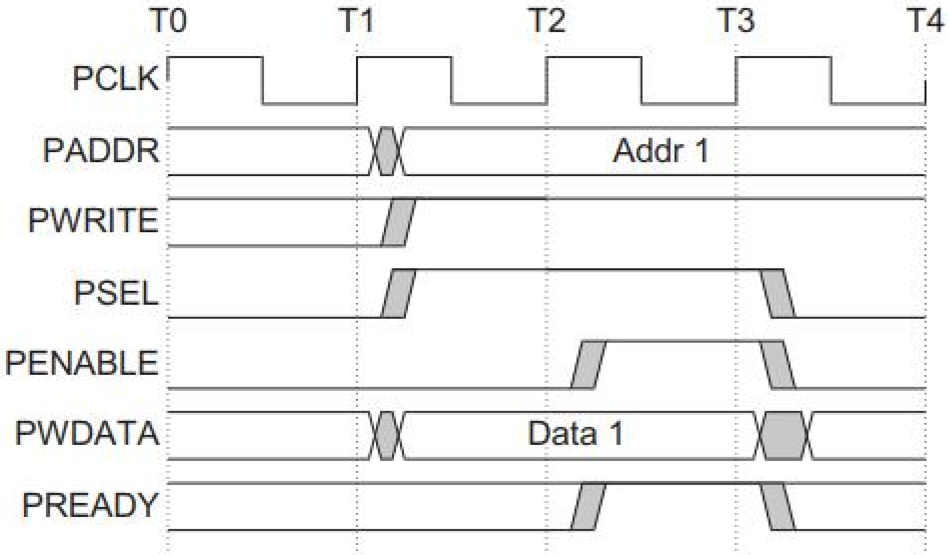


Figure 1: Write Transfer with no wait

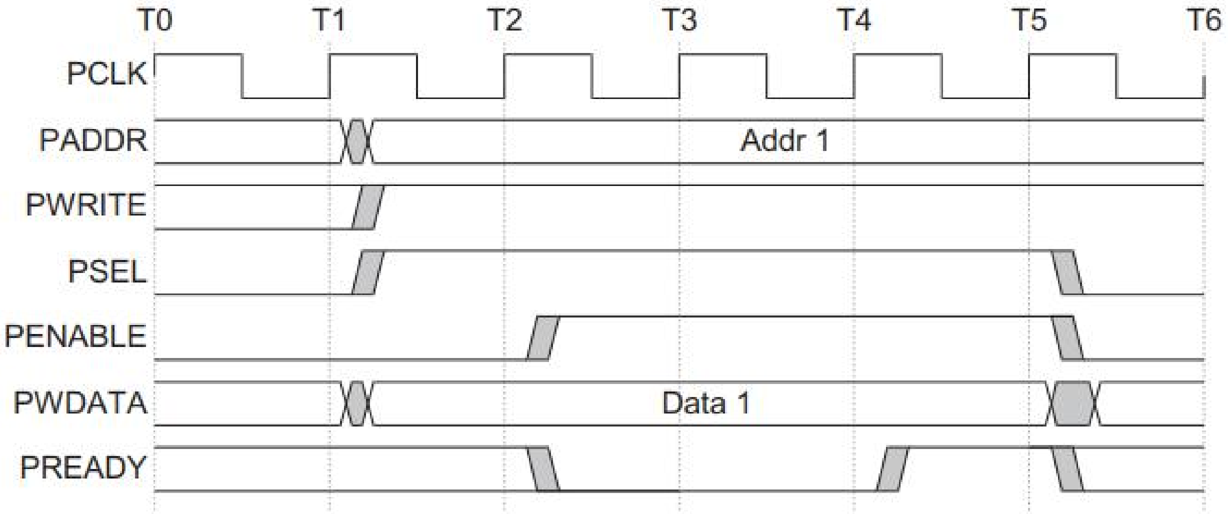


Figure 2: Write Transfer with wait state

3.2.5 Read Operation

The read operation is performed according to the procedure defined in ARM_AMBA3_APB.pdf

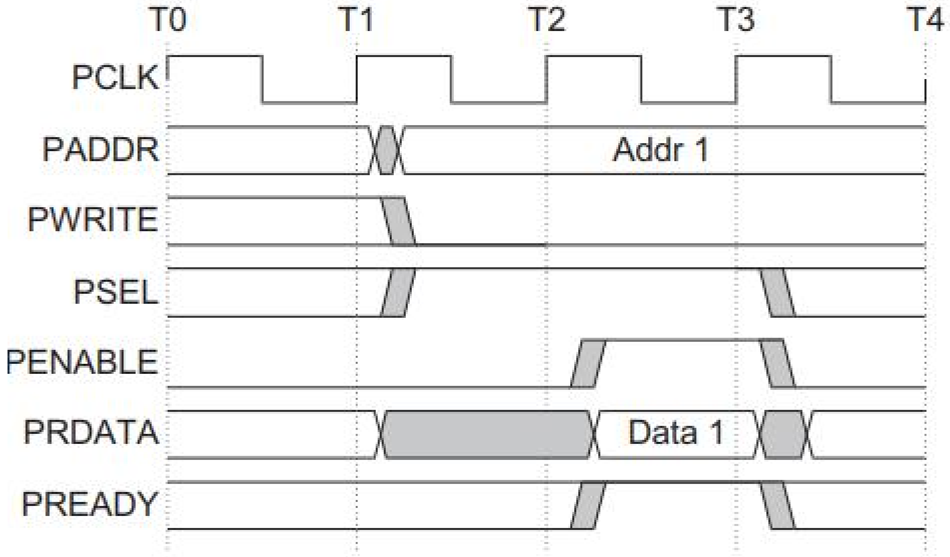


Figure 3: Read Transfer with no wait state

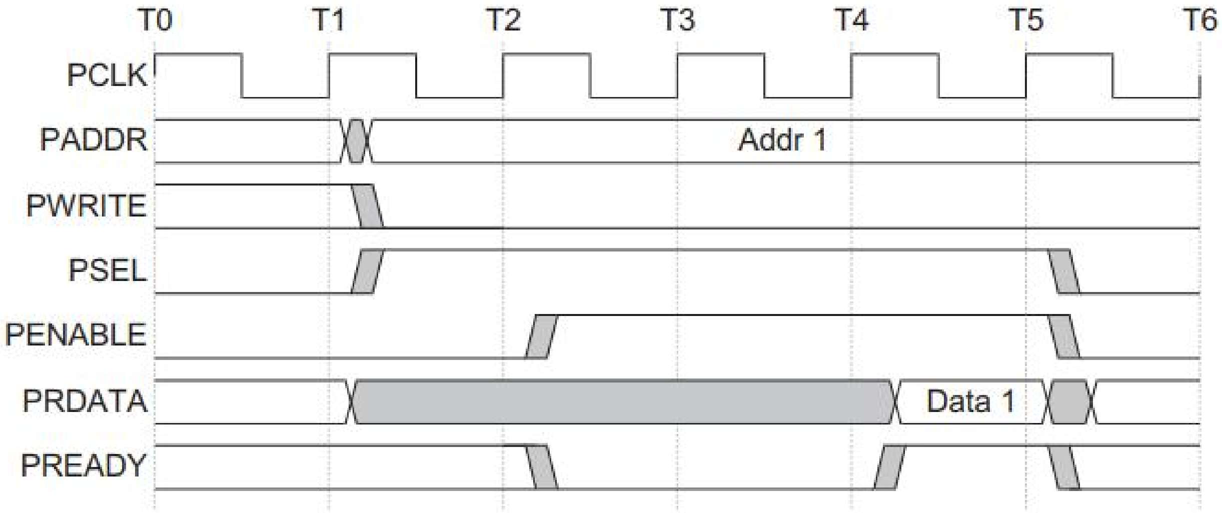


Figure 4: Read Transfer with wait state

3.2.6 Error Response

When SRAM core receives a memory read or write request that is out of address bound it responds with an error by setting PSLVERR signal to 1.

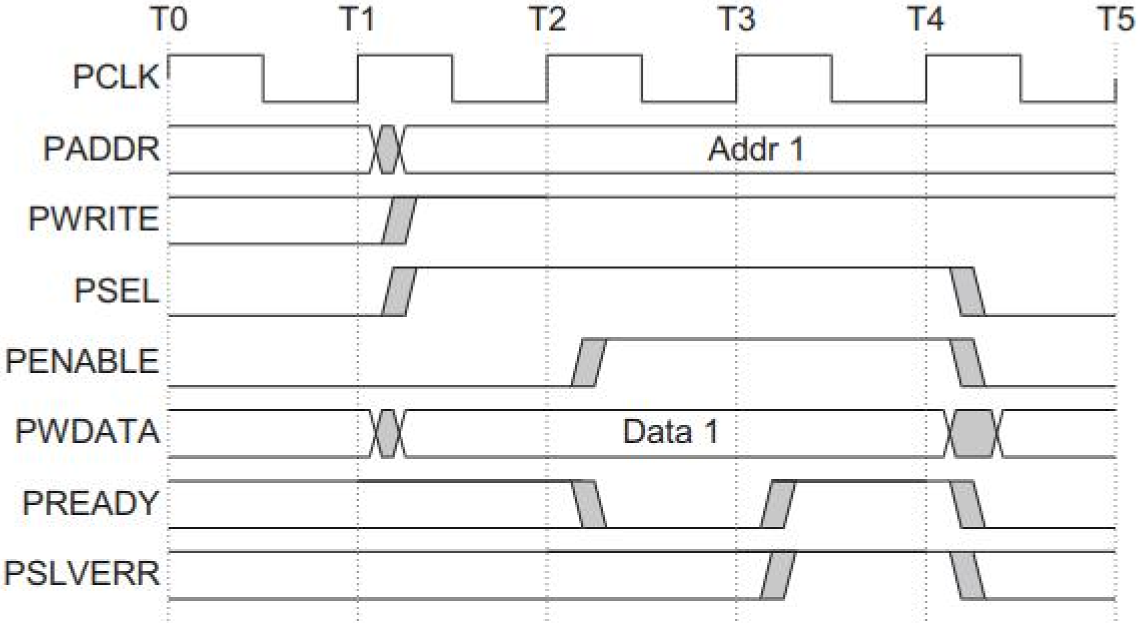


Figure 5: Write Transfer failing

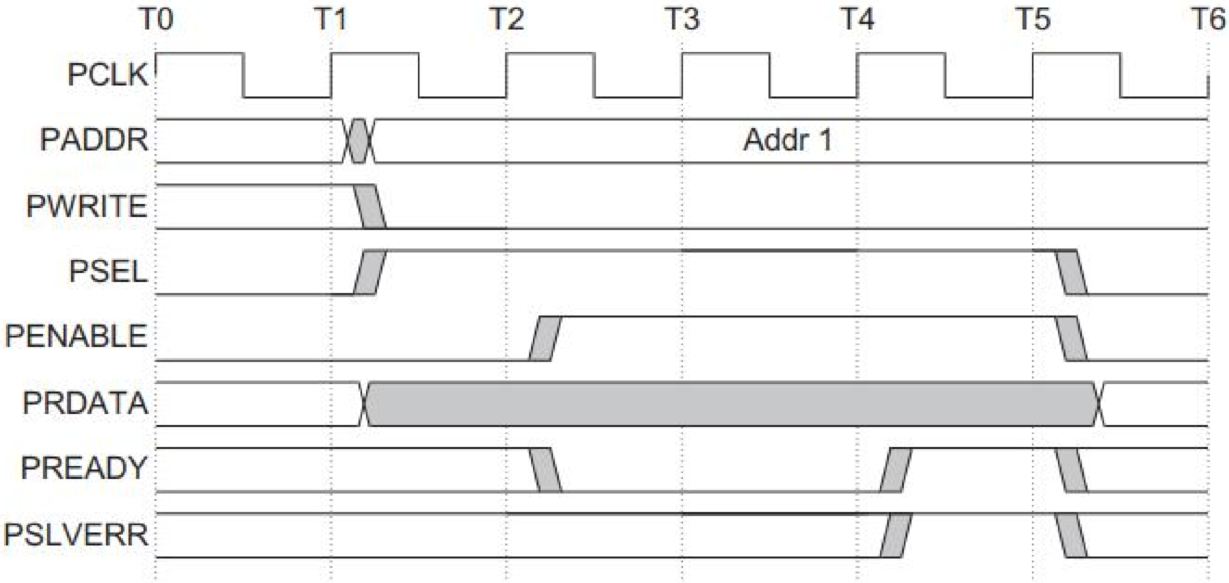


Figure 6: Failing Read Transfer

3.3 SOFTWARE ARCHITECTURE

No additional software or firmware is needed for the SRAM core's operation

3.4 PERFORMANCE

No performance test analysis is done for this core as it is not synthesized yet.

4 SYSTEM DESIGN

4.1 USE-CASES

The APB Slave core can be integrated with multiple designs. The following are some of the user cases,

4.1.1 With APB MUX core

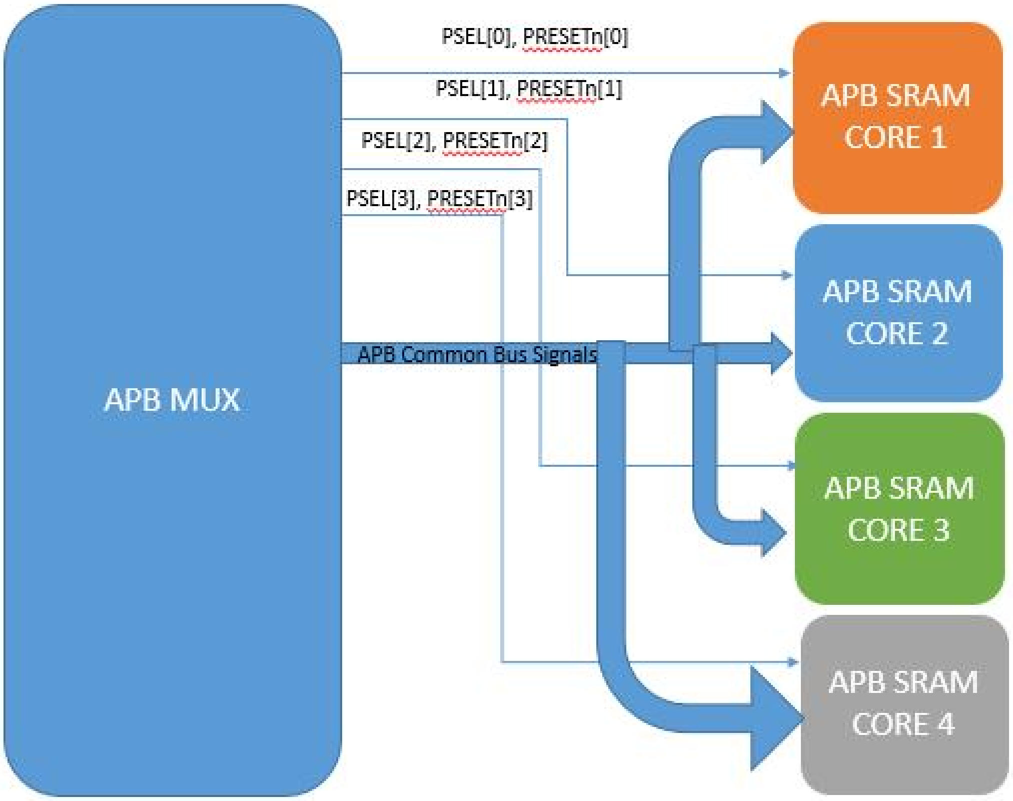


Figure 7: APB SRAM Core with APB MUX Core

4.1.2 With AHB APB Bridge core

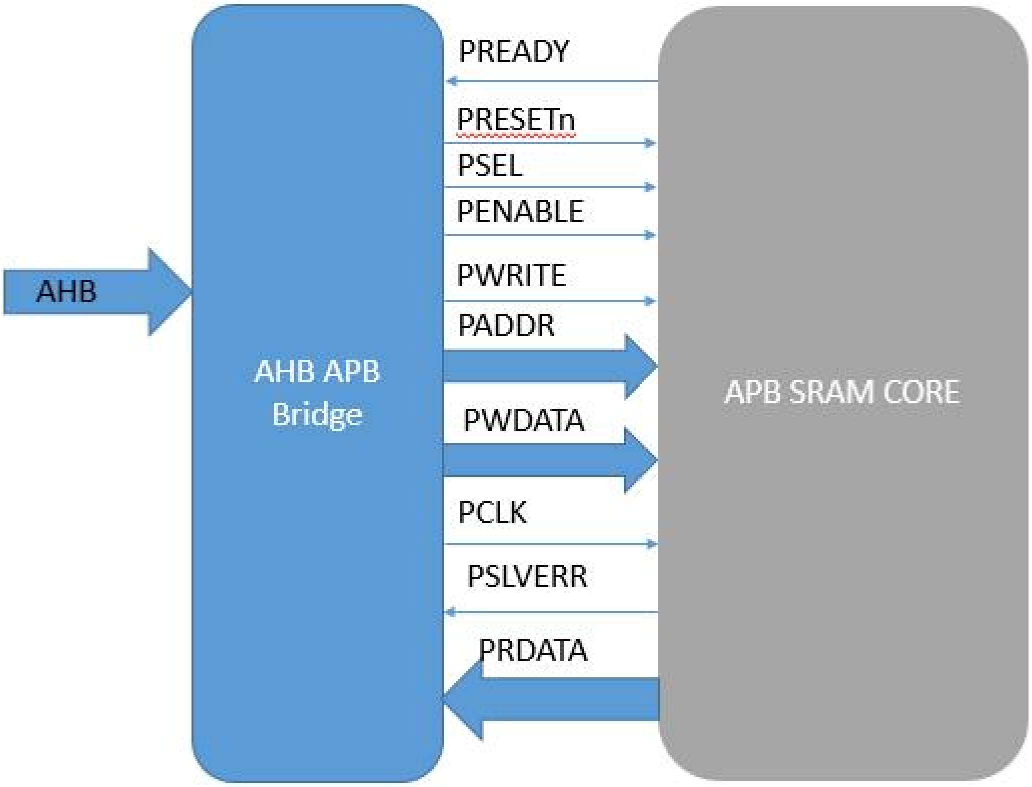


Figure 8: APB_SRAM core with AHB APB Bridge Core

Appendix A: References

The following table summarizes the documents referenced in this document.

Document Name and Version	Description	Location
ARM_AMBA3_APB.pdf	*AMBA v3 APB v1 specification from ARM*	*http://web.eecs.umich.edu/~prabal/teaching/eecs373-f12/readings/ARM_AMBA3_APB.pdf*