DATASHEET

Synchronous Serial Interface (SPI)

Arasan Chip Systems, Inc.
2010 North First Street, Suite #510
San Jose, CA 95131
Ph:408-282-1600
Fx:408-282-7800
www.arasan.com

June, 2018

DISCLAIMER

This document is written in good faith with the intent to assist the readers in the use of the product. Circuit diagrams and other information relating to Arasan Chip Systems' products are included as a means of illustrating typical applications. Although the information has been checked and is believed to be accurate, no responsibility is assumed for inaccuracies. Information contained in this document is subject to continuous improvements and developments.

Arasan Chip Systems' products are not designed, intended, authorized or warranted for use in any life support or other application where product failure could cause or contribute to personal injury or severe property damage. Any and all such uses without prior written approval of an Officer of Arasan Chip Systems Inc. will be fully at the risk of the customer.
Arasan Chip Systems Inc. disclaims and excludes any and all warranties, including without limitation any and all implied warranties of merchantability, fitness for a particular purpose, title, and infringement and the like, and any and all warranties arising from any course or dealing or usage of trade.

This document may not be copied, reproduced, or transmitted to others in any manner. Nor may any use of information in this document be made, except for the specific purposes for which it is transmitted to the recipient, without the prior written consent of Arasan Chip Systems, Inc. This specification is subject to change at any time without notice. Arasan Chip Systems Inc. is not responsible for any errors contained herein.

In no event shall Arasan Chip Systems Inc. be liable for any direct, indirect, incidental, special, punitive, or consequential damages; or for lost of data, profits, savings or revenues of any kind; regardless of the form of action, whether based on contract; tort; negligence of Arasan Chip Systems or others; strict liability; breach of warranty; or otherwise; whether or not any remedy of buyers is held to have failed of its essential purpose, and whether or not Arasan Chip Systems Inc. has been advised of the possibility of such damages.

Restricted Rights
Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor.

Copyright Notice
No part of this specification may be reproduced in any form or means, without the prior written consent of Arasan Chip Systems, Inc.

Questions or comments may be directed to:
Arasan Chip Systems, Inc.
2010 North First Street
Suite 510, San Jose, CA 95131.
Ph: 408-282-1600
Fx: 408-282-7800
Email: sales@arasan.com
http://www.arasan.com

CONTENTS

1 INTRODUCTION 3
1.1 Overview 3
1.2 Features 4
1.2.1 SPI Bus 4
1.2.2 AHB Bus 4
2 ARCHITECTURE 5
2.1 AHB Target/Master Interface 5
2.2 Registers 5
2.3 Clock Pre-scalar 6
2.4 SPI Transceiver 6
2.5 Transmit FIFO 6
2.6 Receive FIFO 6
2.7 Status and Interrupt Generator 6
2.8 DMA (AHB Master Interface) 6
3 PIN DIAGRAM 7
4 SIGNAL INTERFACES 8
5 SoC LEVEL INTEGRATION 11
5.1 Verification Environment 11
5.2 IP Deliverables 11
6 RELATED PRODUCTS 11

TABLES

Table 1: SPI Pins 8
Table 2: AHB Target Interface Pins 9
Table 3: AHB Master Interface Pins 9
Table 4: External Dual Port Memory Pins 10
Table 5: Enable Signals 10
Table 6: Scan Signals 10

FIGURES

Figure 1: SPI Interface Core IP Block Diagram 5
Figure 2: SPI PIN Diagram 7
Figure 3: SPI Verification Environment 11

1 INTRODUCTION

The Synchronous Serial Interface (SPI) is a block that connects to the AHB bus. The AHB _SPI Bridge interfaces to the AHB bus on the system side and the SPI bus on the peripheral side. The AHB interface is used to easily integrate the Bridge Controller for any SOC implementation.

The AHB-SPI is a master/slave interface that enables synchronous serial communication with other master /slave peripherals having a SPI-compatible interface. The controller performs the following functions:

- Parallel-to-serial conversion on data written to an internal 32-bit wide, 32 deep transmit FIFO
- Serial-to-parallel conversion on received data, buffering it in a similar 32-bit wide, 32 deep receive FIFO

Interrupts are generated to during transmit and receive FIFO requests, FIFO overruns during programmed I/O mode, and upon completion of transfers during DMA mode. AHB is a new generation of AMBA bus, which is intended to address the requirements of high performance designs.

AMBA AHB implements the features required for high performance, high clock frequency systems including burst transfers, single clock edge operation and so on. Before an AMBA AHB transfer can commence the bus master must be granted access to the bus. This process is started by the master asserting a request signal to the arbiter. Then the arbiter indicates when the master will be granted use of the bus. A granted bus master starts an AMBA AHB transfer by driving the address and control signals.

1.1 Overview

The SPI – AHB bridge enables an AHB host to access a serial device at high-speed through the SPI interface. The controller can be used in applications such as flash memory card and digital camera. Both AHB and SPI support master and slave modes. The AHB – SPI bridge performs either parallel-to-serial conversion or serial-to-parallel conversion with a maximum throughput of 50 Mbit/sec. A 32-bit x 32-bit transmit FIFO and a 32- bit x 32-bit receive FIFO serve as the data buffer to coordinate data flows between the AHB and SPI interfaces. The AHB master consists of a DMA controller to enhance the system performance.

A SPI Clock Generator is included to provide adjustable input clock to the SPI controller. A SPI clock frequency from 500 KHz to 50 MHz can be selected. The Status and Interrupt Generator provides data transaction information to the AHB host processor that reflects the FIFOs and DMA states. The SPI controller consists of one SPI master and one SPI slave and it can be programmed by an AHB host to support the TI, Motorola, or National SPI protocol. Full SPI duplex mode is supported. The Arasan High Speed SPI – AHB IP Core is

an RTL design in Verilog that implements an SPI – AHB controller on an ASIC, or FPGA. The Arasan High Speed SPI – AHB IP Core has been widely used in different applications by major chip vendors.

1.2 Features

1.2.1 SPI Bus

- Compliant with Motorola SPI specification
- Compliant with TI SPI specification
- Compliant with National SPI specification
- Supports both SPI master and SPI slave Operations
- Maximum 50 Mbit/sec data throughput
- Pre-scaling for programmable clock rate from 500 KHz to 50 MHz
- Supports full-duplex mode

1.2.2 AHB Bus

- Compliant with AMBA revision 2.0 specification
- Supports AHB bus for varying frequency range from 1 MHz to 100MHz
- Supports Bus master DMA modes
- Supports both AHB master and AHB slave modes
- Supports interrupts
- 32 x 32 bytes transmit and receive FIFOs for high performance AHB - SPI data transfers

2 ARCHITECTURE

The figure below displays AHB-SPI Bridge Controller block diagram. The block has an AHB target interface to read and write registers and to perform programmed IO transfers. The AHB master interface is used to generate DMA cycles for transferring data to and from memory during DMA transfers. The SPI interface handles the serializing and de-serializing of data to the SPI interface.

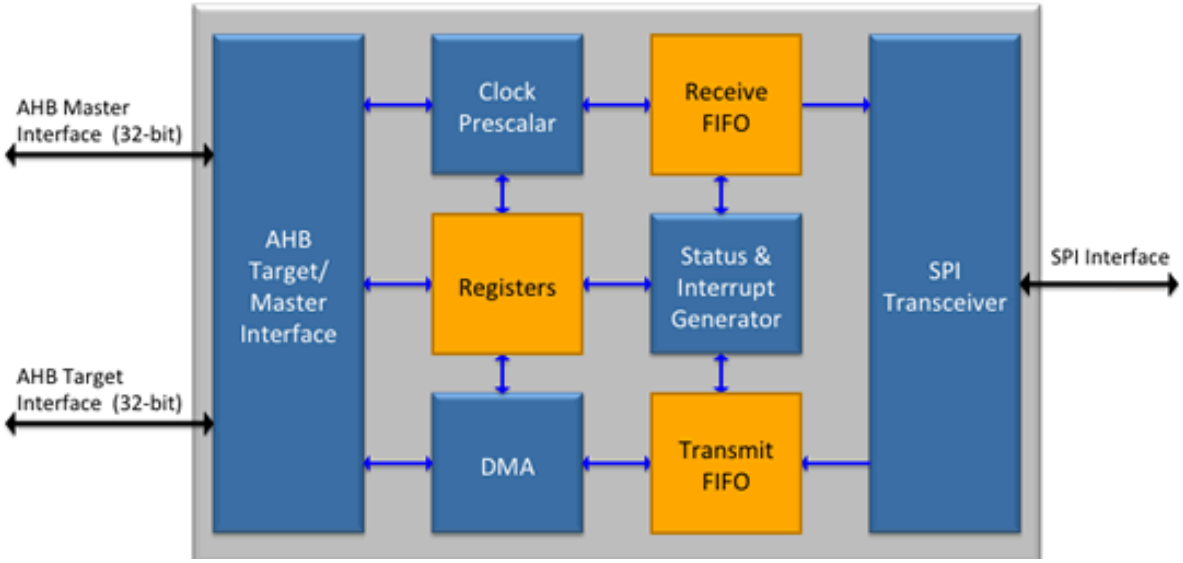

Figure 1: SPI Interface Core IP Block Diagram

2.1 AHB Target/Master Interface

The AHB target interface is primarily used for programming the registers on the SPI controller and also to transfer data is the PIO mode is used (DMA mode disabled).

2.2 Registers

The target registers are used to program the controller and also to give the status of the design. The registers are programmed using the AHB interface and are used by the SPI controller to control the SPI interface. When necessary the signals are synchronized to the SPI clock domain.

2.3 Clock Pre-scalar

The clock prescalar contains a running counter. The counter is used to generated a divide signal and generate the SPI clock as required by the user.

2.4 SPI Transceiver

The SPI transmit and receive logic contains the state machines.

2.5 Transmit FIFO

This FIFO is 32 bit and 32 deep. It is filled by the DMA controller and read by the SPI controller during a transmit operation.

2.6 Receive FIFO

This FIFO is 32 bit and 32 deep. If is filled by the SPI controller and read by the DMA controller during a receive operation.

2.7 Status and Interrupt Generator

The interrupts are generated for the operations described in the interrupt registers.

2.8 DMA (AHB Master Interface)

The DMA controller reads data from memory and puts it into the transmit FIFO which then sent out on the SPI interface. Similarly, it gathers data from the SPI interface in the receive FIFO and dumps into the ARM processor during read operations from SPI interface. DMA controller is used in both SPI master and SPI slave modes of operations.

3 PIN DIAGRAM

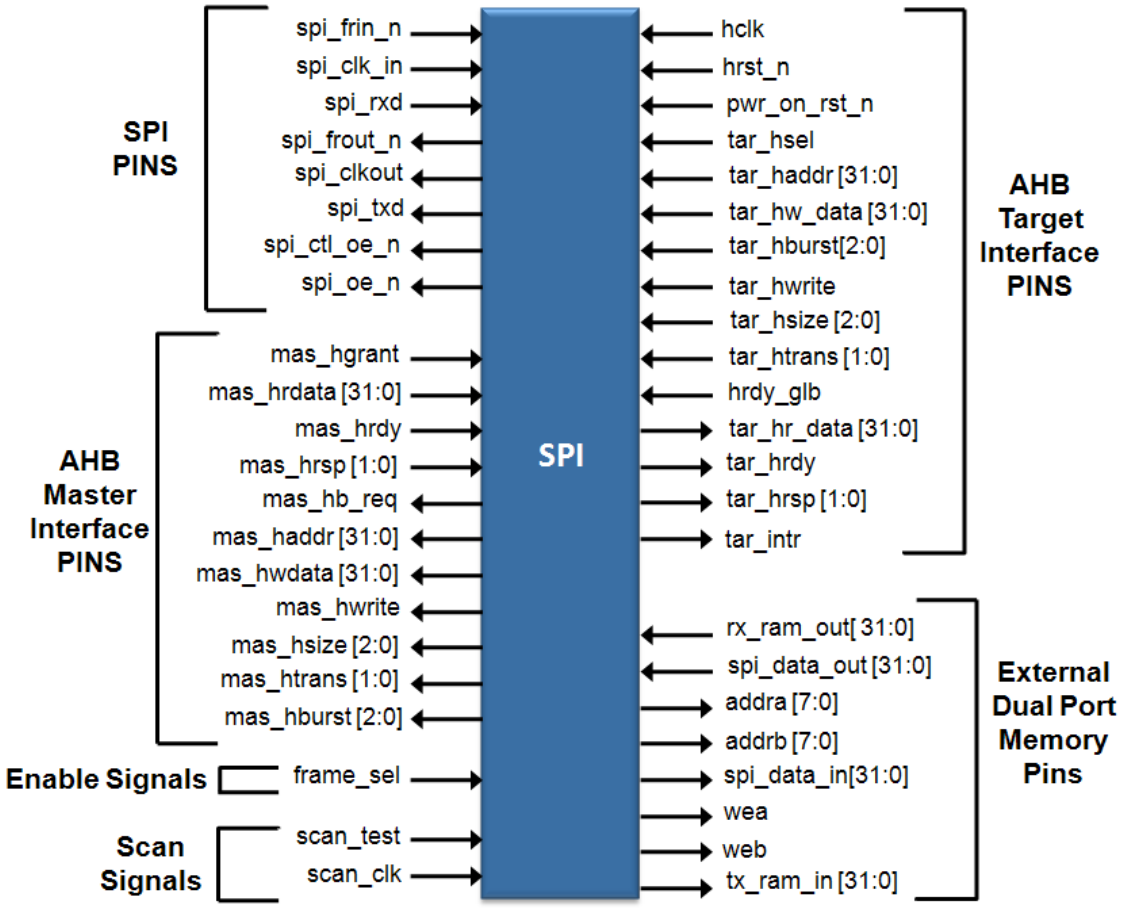

Figure 2: SPI PIN Diagram

4 SIGNAL INTERFACES

The SPI describes signal names and its descriptions used in high speed SPI bus interface

- SPI Pins
- AHB target interface signals
- AHB master interface signals
- External dual port memory signals
- Enable signals
- Scan signals

Table 1: SPI Pins

Pin Name	Direction	Description
spi_frin_n	Input	SPI Frame Signal input when configured as a slave. This signal is low when active and high when inactive
spi_clk_in	Input	This input signal is the SPI clock used by the slave when transmitting or receiving data
spi_rxd	Input	Receive data input. This data is put into the FIFO by the SPI controller
spi_frout_n	Output	SPI Frame Signal to select the slave. This signal is low when active and high when inactive
spi_clkout	Output	This output signal is the SPI clock generated by the master when transmitting or receiving data
spi_txd	Output	Transmit data output. This data is sent from the FIFO
spi_ctl_oe_n	Output	SPI Clock Output Enable: This signal is used for the clock. This signal is cleared when the device is in master mode and set when it is in slave mode
spi_oe_n	Output	SPI Output Enable: This signal controls the bi-directional pad when the spi_txd is used a bidirectional signal for transmitting and receiving data.

Table 2: AHB Target Interface Pins

Pin Name	Direction	Description
hclk	Input	AHB Clock. This is the main clock to the SPI module. The SPI clock is derived from this clock depending the programming of the divisor values in the registers.
hrst_n	Input	AHB Reset: Active low reset received from the AHB interface
pwr_on_rst_n	Input	Active low power on reset signal
tar_hsel	Input	Slave Select
tar_haddr [31:0]	Input	Address Bus (Byte Addresses)
tar_hw_data [31:0]	Input	Write Data Bus
tar_hburst[2:0]	Input	Burst Size
tar_hwrite	Input	Write or Read Direction Indication
tar_hsize [2:0]	Input	Size (Byte, Half Word or Word)
tar_htrans [1:0]	Input	Transfer Type
hrdy_glb	Input	Global hready
tar_hr_data [31:0]	Output	Read Data Bus
tar_hrdy	Output	Target Read
tar_hrsp [1:0]	Output	Transfer Response
tar_intr	Output	Interrupt to the ARM

Table 3: AHB Master Interface Pins

Pin Name	Direction	Description
mas_hgrant	Input	AHB Bus Grant
mas_hrdata [31:0]	Input	Read Data
mas_hrdy	Input	Target Ready
mas_hrsp [1:0]	Input	Transfer Response
mas_hb_req	Output	AHB Bus Request
mas_haddr [31:0]	Output	Bus Address (Byte Address)
mas_hwdata [31:0]	Output	Write Data
mas_hwrite	Output	Write or Read Direction Indication
mas_hsize [2:0]	Output	Size (Byte, Half Word or Word)
mas_htrans [1:0]	Output	Transfer Type
mas_hburst [2:0]	Output	Burst Size

Table 4: External Dual Port Memory Pins

Pin Name	Direction	Description
rx_ram_out[31:0]	Input	Read data for AHB interface
spi_data_out[31:0]	Input	Read data for SPI interface
addra[7:0]	Output	Write and read address for SPI
addrb[7:0]	Output	Write and read address for AHB
spi_data_in[31:0]	Output	Data input from SPI interface
wea	Output	Write enable signal for SPI interface
web	Output	Write enable signal for AHB interface
tx_ram_in[31:0]	Output	Input data from AHB memory

Table 5: Enable Signals

Pin Name	Direction	Description
frame_sel	Output	Enable signal to assign spi_frout_n

Table 6: Scan Signals

Pin Name	Direction	Description
scan_test	Input	Enable signal for scan operation
scan_clk	Input	Scan clock for scan operation

5 SoC LEVEL INTEGRATION

5.1 Verification Environment

This section displays the RTL verification and test environment.

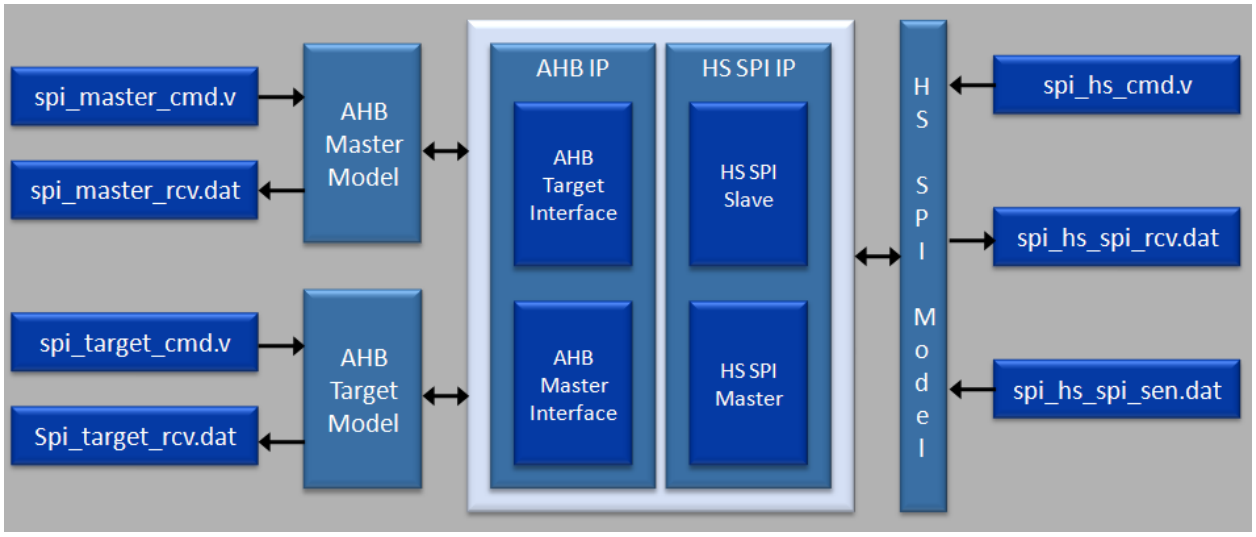

Figure 3: SPI Verification Environment

5.2 IP Deliverables

The IP package consists of the following:

- RMM-compliant synthesizable RTL design in Verilog
- Easy-to-use test environment
- Synthesis scripts
- Technical documents
- Simulation scripts

6 RELATED PRODUCTS

- I2S Controller
- I2C Controller