

SoCLabs XiP AHB QSPI Configuration and User Manual

Daniel Newbrook, SoC Labs

January 26, 2025

Contents

1	Introduction	2
1.1	IP	2
1.1.1	Arm(R)	2
1.2	Integrating this IP	2
2	Programmers Manual	4
2.1	Address Map	4
2.2	Setting XiP Mode	4
2.3	Write to Flash	5

Chapter 1

Introduction

In SoC designs it is often quite valuable to have external non-volatile memory. Typically this will be where instruction code can be stored, and ideally we don't want to have to program this memory space after every power cycle.

QSPI flash is fairly inexpensive and the protocol is fairly simple to implement. Whilst it isn't the fastest protocol that you could use, it also doesn't require any real effort in the analog domain unlike for example SATA.

Execute in place (XiP) is the method of making an external flash look like a memory mapped region to the processor. Meaning the processor can directly run instructions from this address space. We add a small cache between the QSPI controller and the bus to reduce access time for repeat accesses.

The design here uses an AHB interface for the main XiP interface, and an APB interface for access to internal registers of this IP, as well as to send commands over the QSPI interface.

1.1 IP

The SoCLabs AHB QSPI relies on IP's from certain vendors. In order to build you will need the following IP.

1.1.1 Arm(R)

Arm IP should be downloaded from Arm and placed into the recommended IP directories.

- Corstone-101
- CG092 (flash cache)

1.2 Integrating this IP

There are 2 steps to integrating this IP in an SoC

1. Include the IP in the file list
2. Instantiate the IP in the design

For step 1. you make an environment variable at your top level project pointing to the ahb qspi directory called SOCLABS_AHB_QSPI_DIR (this will help with path references). You should then include the ahb_QSPI_SIM.flist to your behavioural flist, and ahb_QSPI_ASIC.flist to your asic flist.

For step 2. you need an APB and AHB port to connect to, and then to bring out the QSPI wires to the top of your design. I think connecting these to simple tristate pads should be fine.

Chapter 2

Programmers Manual

2.1 Address Map

The APB interface can access a number of registers. Outlined below are only the ones relating to the developed IP here, for the CG092 register please see documentation from Arm.

Table 2.1: Register Address Map

Register	Address
Control Reg	0x0000
Status Reg	0x0004
SPI Commands	0x0008
SPI Address	0x000C
Read Data 0	0x0010
Read Data 1	0x0014
Read Data 2	0x0018
Read Data 3	0x001C
Write Data 0	0x0020
Write Data 1	0x0024
Write Data 2	0x0028
Write Data 3	0x002C
CG092 Cache control	0x1000-0x1FFF

2.2 Setting XiP Mode

Typical flow for entering XiP

1. Reset the QSPI flash

Table 2.2: Control Register Bits

Bits	Function
0	QSPI Quad IO Mode
8	XiP Active
16-23	QSPI Mode Code
24	Continous Read Active
25	No CMD active

Table 2.3: Status Register Bits

Bits	Function
0	QSPI Busy

Table 2.4: SPI Commands Register Bits

Bits	Function
0-7	QSPI Command
8	QSPI Enable
9	QSPI Read Enable
10	QSPI Write Enable
11	QSPI Address enable
12-15	Number Dummy cycles (N-1)
16-19	Number R/W Bytes (N-1)

Table 2.5: QSPI Address Register Bits

Bits	Function
0-21	QSPI Address

2. Enable QSPI Mode
3. Set Continous Read mode
4. Set XiP/AHB Mode
5. Configure Cache

2.3 Write to Flash

Typical flow for writing to flash

1. Reset the QSPI Flash

2. Enable QSPI mode
3. Clear write protection
4. Set write enable
5. Write to APB registers
6. Write to flash
7. Poll flash status until complete