3.4 Hardware Accelerator InterFAce (HAIFA)

3.4.1 Overview

The HW accelerator interface of the SC1XXS sub-system transfers data between the StarCore DSP core and HW accelerators added outside the subsystem.

Main features of the hardware accelerator interface:

- Data is Written and read as memory mapped
- Simple one master direct interface
- Zero wait statesfor write (two wait states for read)
- Deterministic access time to the accelerator
- Up to 1MBytes of accelerator memory mapped registers
- Address space loaction is hardware configurable

3.4.2 Functional Description

The SC1XXS data bus protocol is the basis for the interface protocol between the accelerator and the StarCore.

The protocol between the accelerator and the HAIFA (Hardware Accelerator InterFAce) is as follows:

The coprocessor registers are mapped on the core 4 GByte memory space. In order to save routing on the subsystem interface, a 64 KByte memory space is dedicated for the accelerators usage only.

The system designer configures the location of the 1MByte accelerator memory space using the gg_haifa_bank_sel input bus (i.e: Those 12 bits are the 12 MS Bits of the physical memory address).

The HAIFA compares both xa_ap_addr and xb_ap_addr address busses bits 31:20 with the gg_haifa_bank_sel bits. The matching bus is used for accessing the HW accelerators Bus. If both StarCore data buses attempt to access the hardware accelerator simultaneously, the HAIFA asserts the xc_dp_m2s_error signal high, indicating an error during the access to the hardware accelerators.

The HAIFA block buffers all signals at the sub-system interface, therefore enabling full speed back to back write access to the Accelerators memory from the core.

Read accesses insert two wait states implicitly by the HAIFA block, reflecting the data read delay due to the address buffering from the HAIFA, and data back buffering to the HAIFA.

If data is further delayed from any reason by the accelerator, then the accelerator must insert a WAIT signal on the expected data cycle.

3.4.3 HAIFA Interfaces Diagram

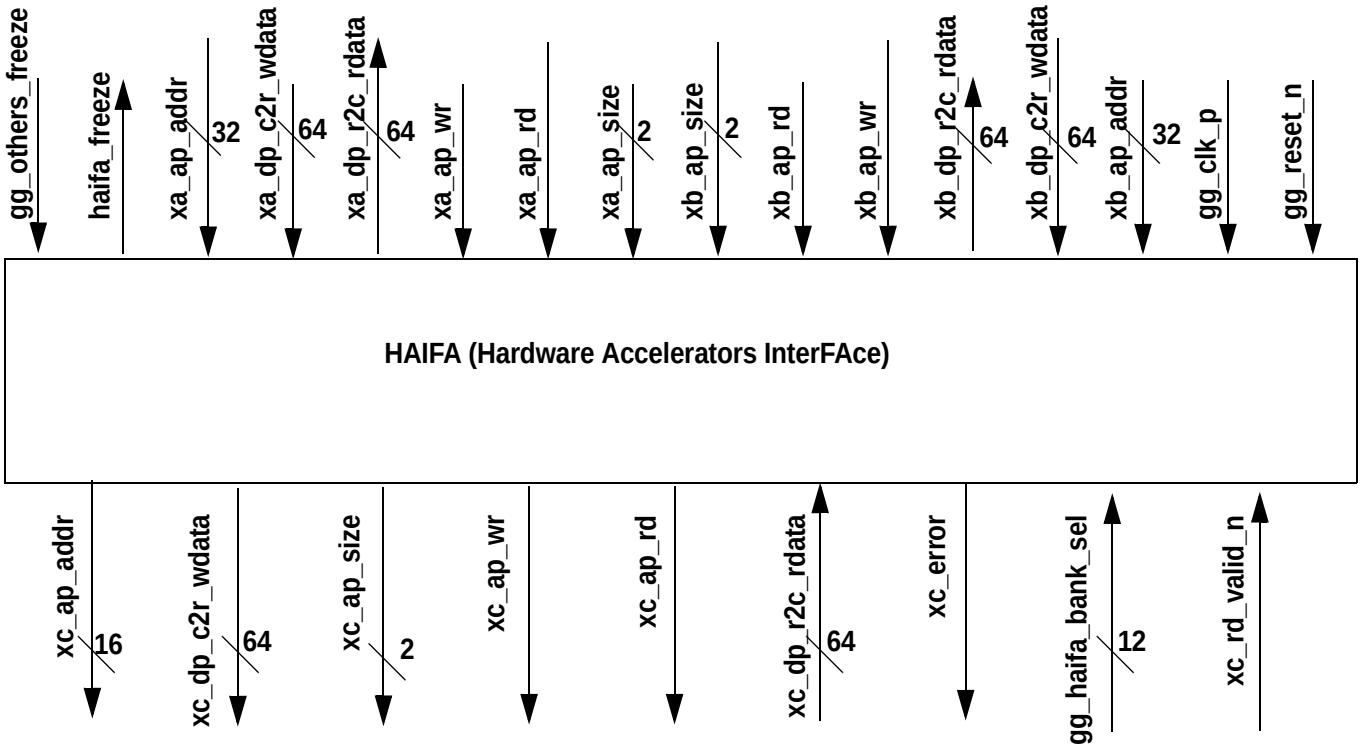

Figure 26: Accelerators Interface Block

3.4.4 HAIFA Pinout List

Table 13: HAIFA Pinout List (Sheet 1 of 2)

Pin	Direction	Width	Description
gg_clk_p	in	1 -bit	Core clock signal
gg_reset_n	in	1 -bit	Core reset signal
xa_ap_addr	in	32 -bit	Core address bus A
xa_dp_c2r_wdata	in	64 -bit	Core write data bus A
xa_dp_r2c_rdata	out	64 -bit	Core read data bus A
xa_ap_rd	in	1 -bit	Core data bus A read enable
xa_ap_wr	in	1 -bit	Core data bus A write enable

Table 13: HAIFA Pinout List (Sheet 2 of 2)

Pin	Direction	Width	Description
xa_ap_size	in	2 -bit	Core data bus A access size Byte - “00” Word (16 -bit) - “01” Long Word (32 -bit) - “10” Two Long Words (64 -bit) - “11”
xb_ap_addr	in	32 -bit	Core address bus B
xb_dp_c2r_wdata	in	64 -bit	Core write data bus B
xb_dp_r2c_rdata	out	64 -bit	Core read data bus B
xb_ap_rd	in	1 -bit	Core data bus B read enable
xb_ap_wr	in	1 -bit	Core data bus B write enable
xb_ap_size	in	2 -bit	Core data bus B access size Byte - “00” Word (16 -bit) - “01” Long Word (32 -bit) - “10” Two Long Words (64 -bit) - “11”
xc_ap_addr	out	20 -bit	Accelerator address bus - either xa_ap_addr or xb_ap_addr -bits [19:0] according to the selected bus
xc_dp_c2r_wdata	out	64 -bit	Accelerator write data bus - either xa_dp_m2s_wdata or xb_dp_m2s_wdata according to the selected bus
xc_dp_r2c_rdata	in	64 -bit	Accelerator read data bus
xc_ap_rd	out	1 -bit	Accelerator data bus read enable
xc_ap_wr	out	1 -bit	Accelerator data bus write enable
xc_ap_size	out	2 -bit	Accelerator data bus access size Byte - “00” Word (16 -bit) - “01” Long Word (32 -bit) - “10” Two Long Words (64 -bit) - “11”
gg_haifa_bank_sel	in	12 -bit	Accelerator memory mapped 1MBytes bank location in the total 4G memory space.
xc_rd_valid_n	in	1 -bit	when ‘1’ - indication to the HAIFA that read data is not ready
xc_error	out	1 -bit	Accelerator memory Access error, inserted whenever there is an access attempt from both memory buses to the coprocessor 1MByte memory area on the same cycle.
haifa_freeze	out	1 -bit	Indicate to the core that read data is not ready.
gg_others_freeze	in	1 -bit	Indicate that the core gg_freeze is driven active by sub-system (core gg_freeze signal excluding haifa_freeze).

3.4.5 HAIFA Timing diagrams

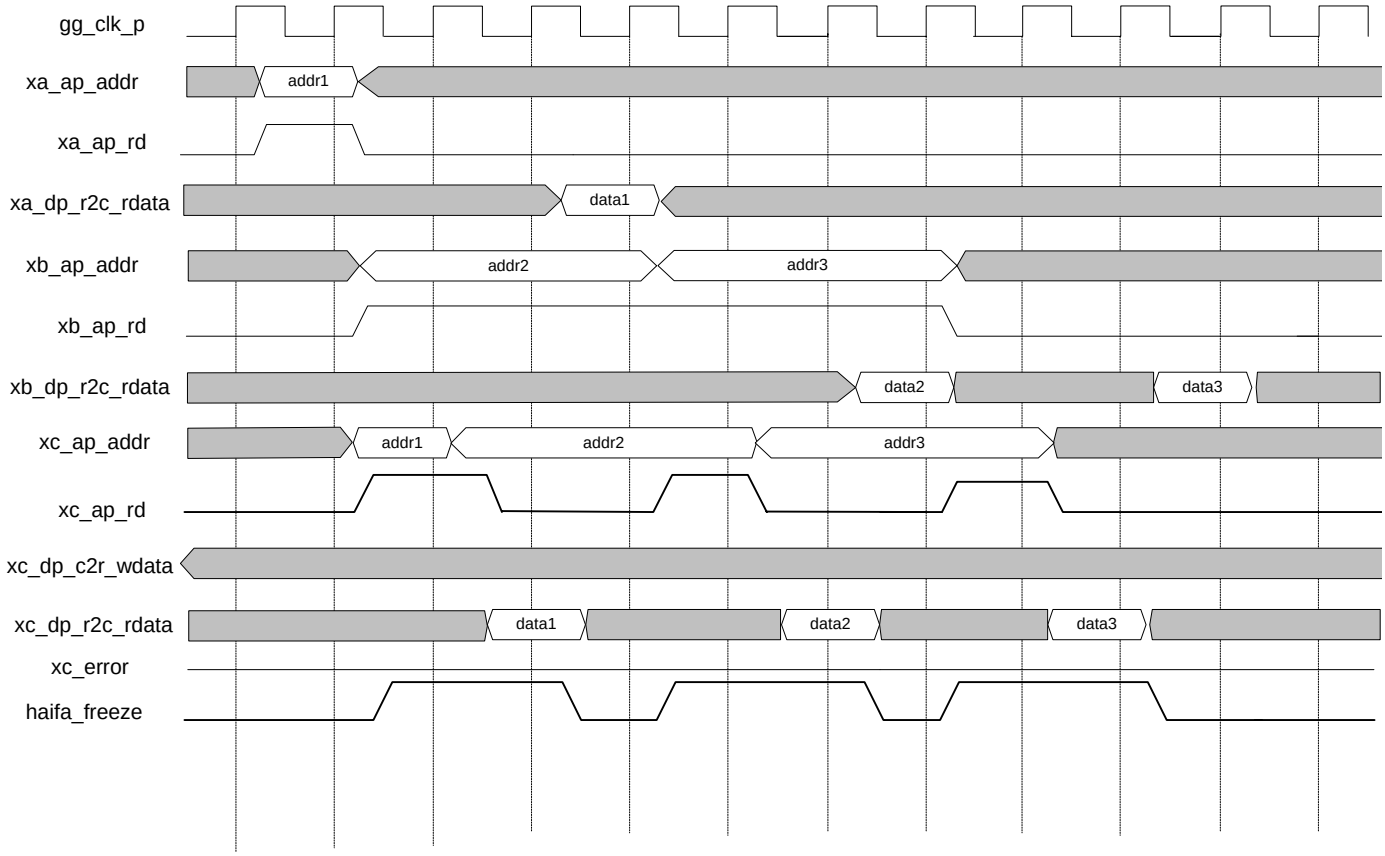

Figure 27: Three Accelerator Read Cycles

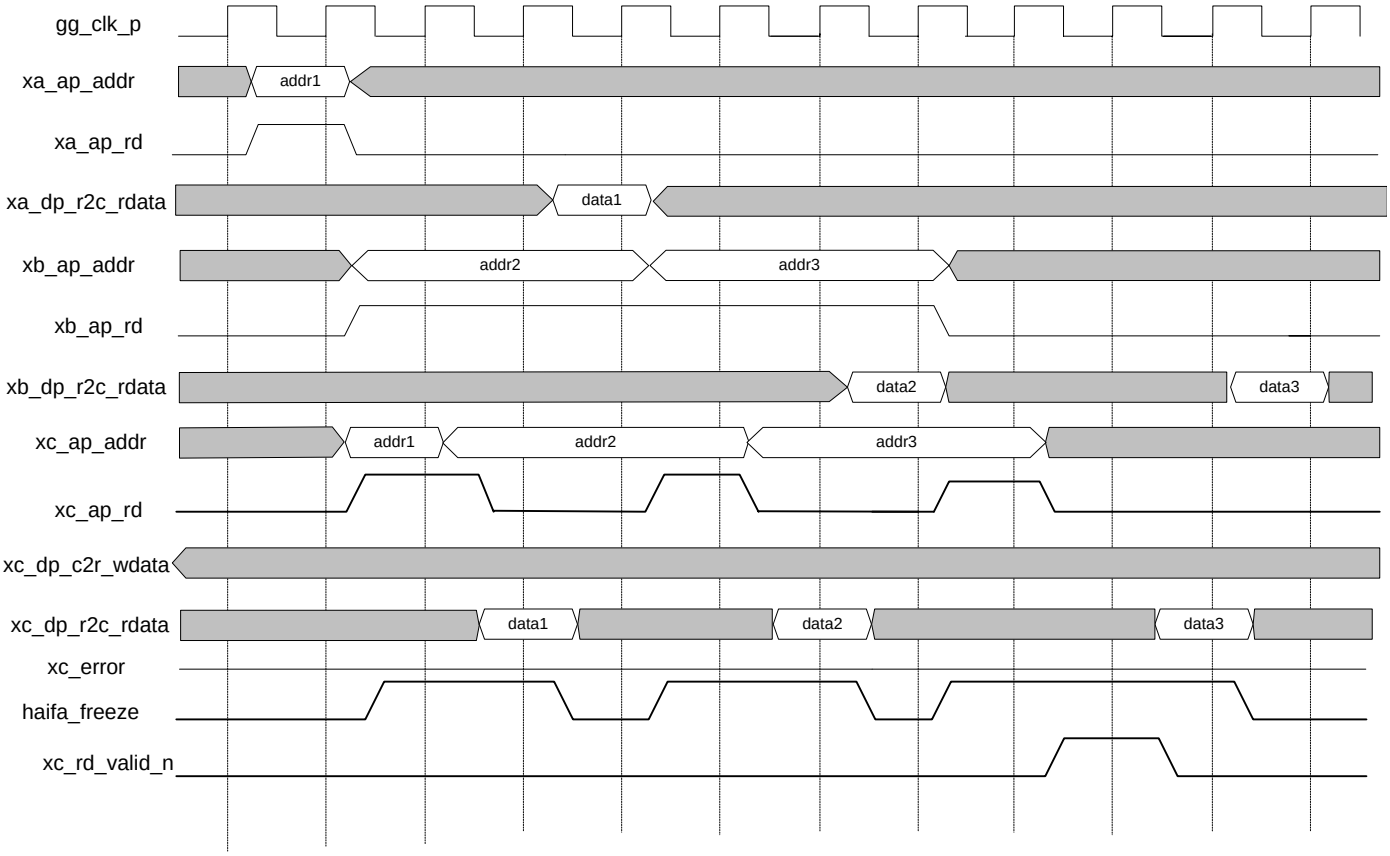

Figure 28: Three Accelerator Read Cycles with One Wait Cycle

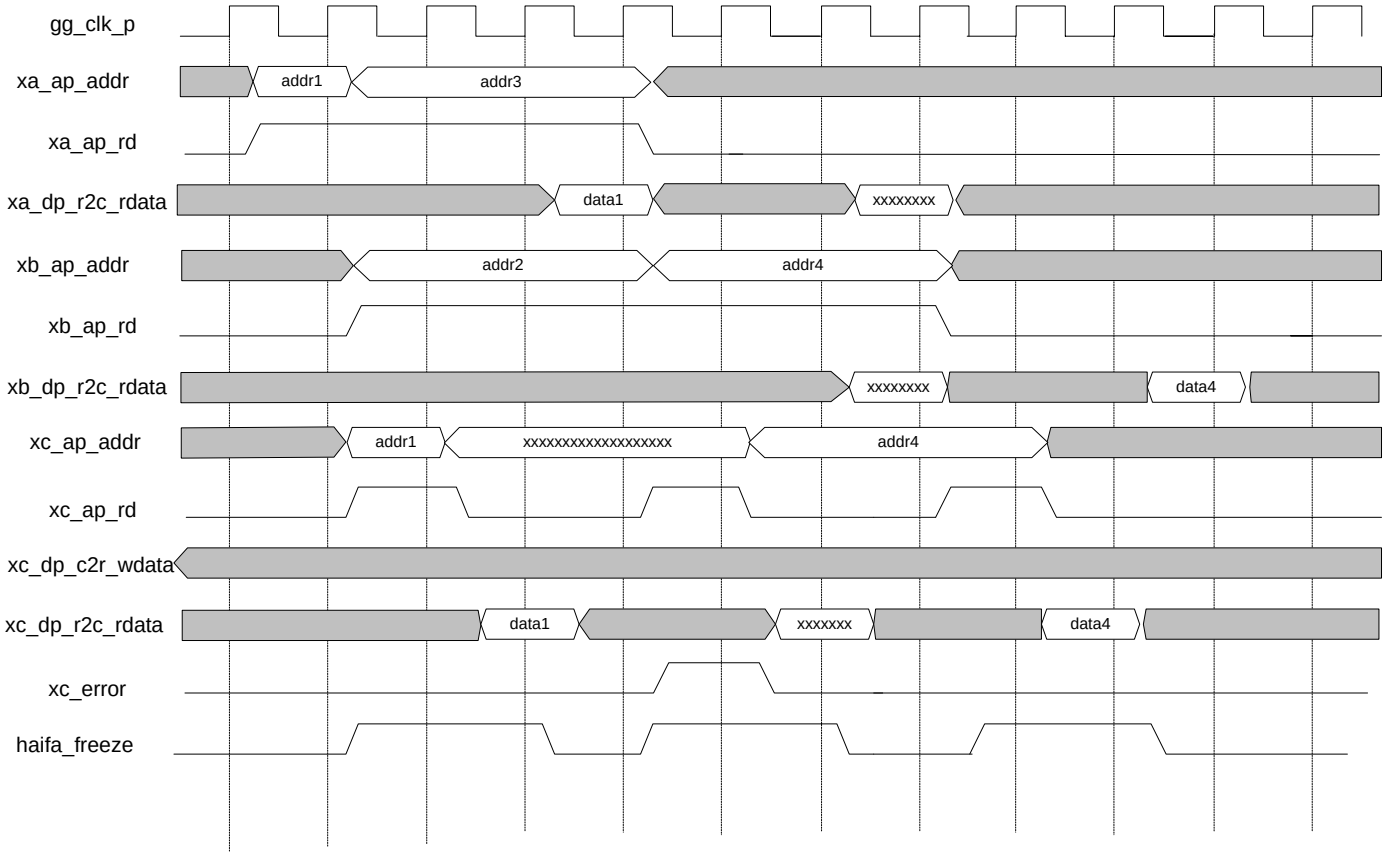

Figure 29: Four Accelerator Read Cycles with One Contention Cycle

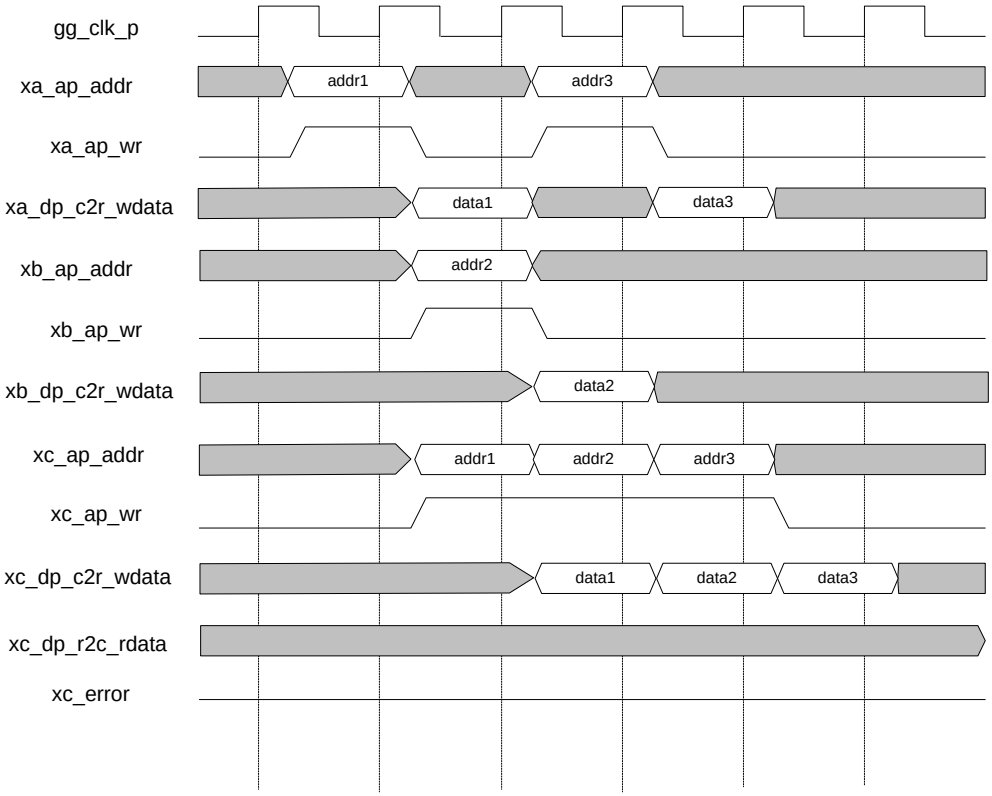

Figure 30: Three Accelerator Write Cycles

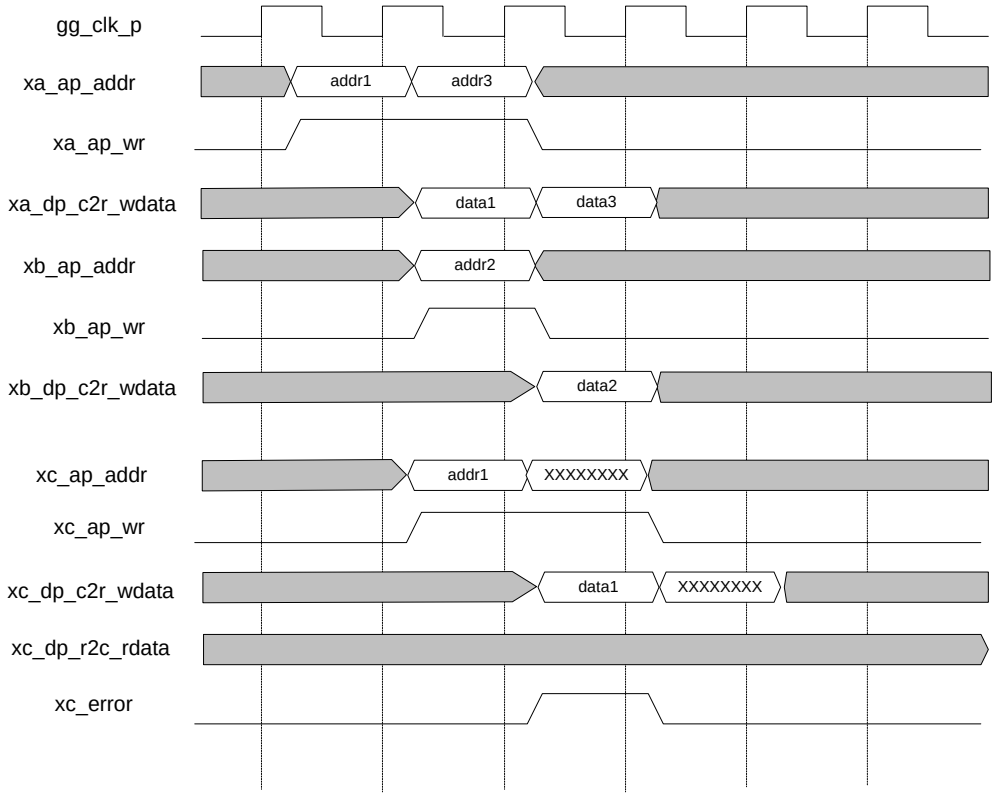

Figure 31: Three Accelerator Write Cycles with One Contention

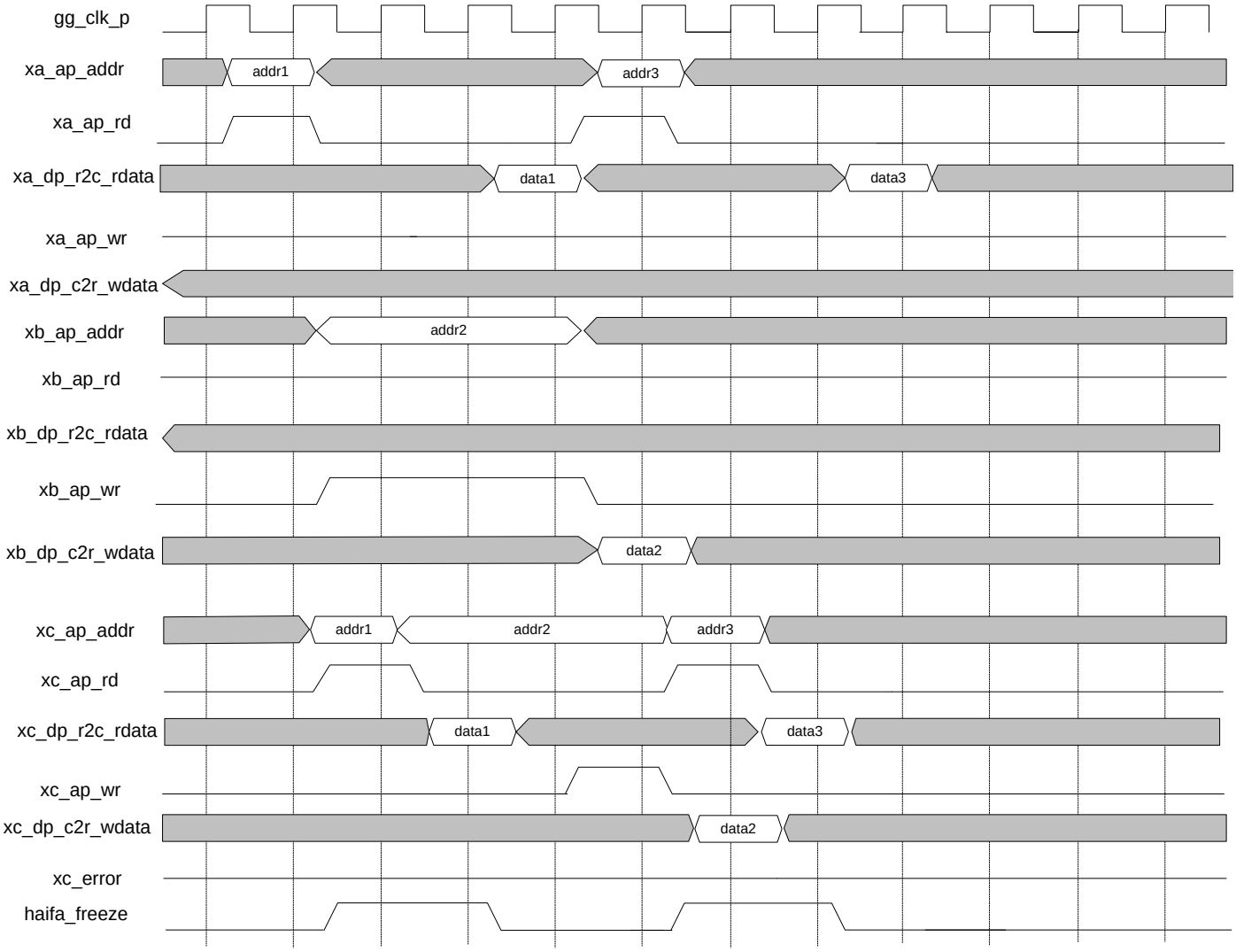

Figure 32: Read Write Read Accelerator Cycles

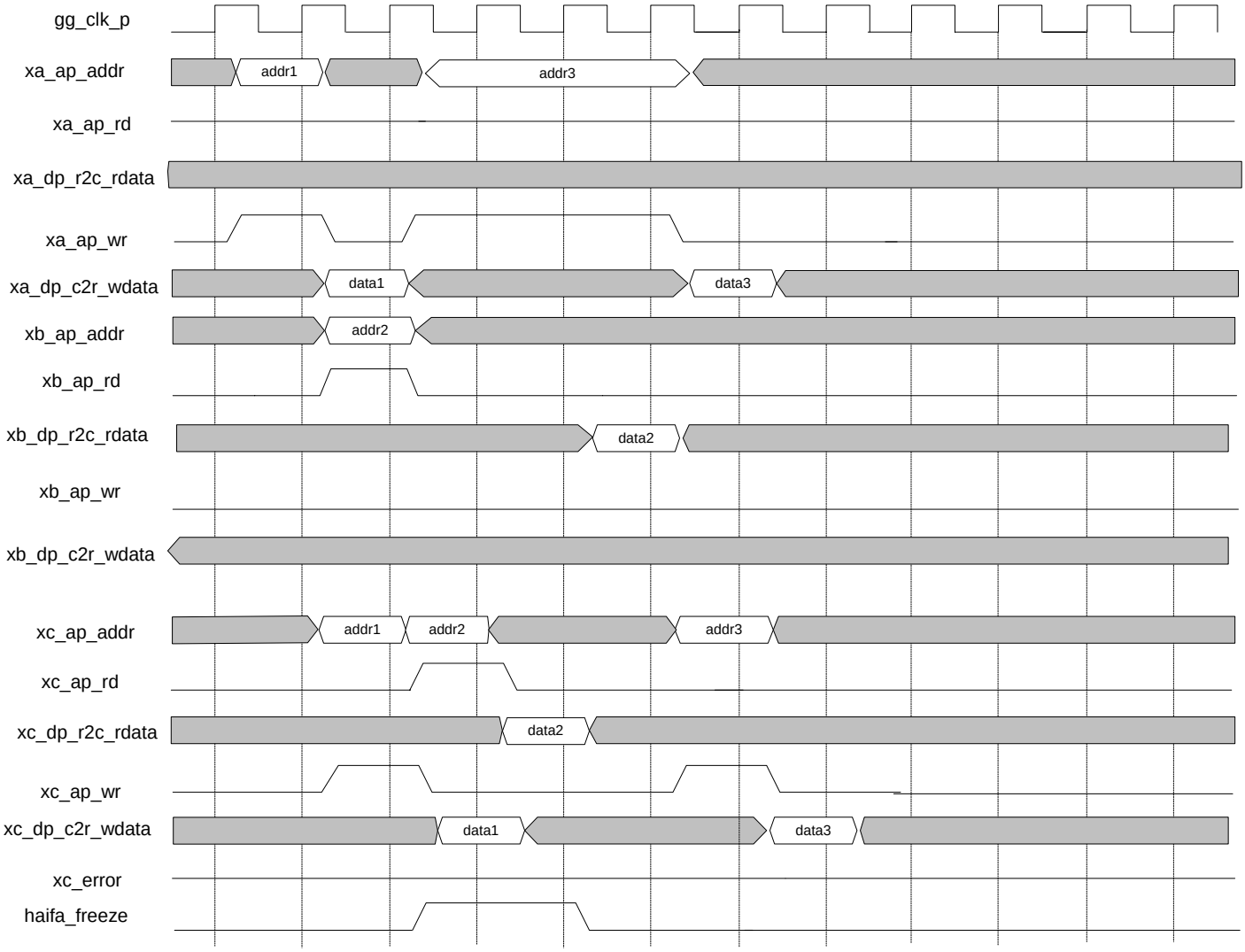

Figure 33: Write Read Write Cycles