European XFEL
SYS_INIT
Data Sheet

Content

Table of Contents

Figures

1 Introduction

For the European XFEL project an initialization interface for the use with the various functionalities and boards was developed. This initialization shall use storage for the settings of the functionalities and boards and follow an initialization sequence and the operation after. The interface is generic enough to fulfill the needs for various systems like Cavity BPMs and Button BPMs:
This component provides a BRAM interface to the storage of the initialization sequence data, a FIFO which stores the address sequence of written data (can by bypassed) and a register which triggers an interrupt to the local processor in order to initiate the initialization.

1.1 Purpose

The document here describes the firmware developed.

1.2 Scope

This document provides a detailed overview of the firmware interface and specifies the user interface.

1.3 Definitions, acronyms, and abbreviations

This document is based on the “IEEE Recommended Practice for Software Requirements Specifications” [1].

ADC	Analog Digital Converter.
BPM	Beam Position Monitor
FPGA	Field Programmable Gate Array. Programmable logic device.

1.4 References

[1] IEEE Std 830-1998, Recommended Practice for Software Requirements Specifications.

1.5 Overview

Chapter 2 provides an overview and how the firmware is related to the other firmware used. Chapter 3 contains all the detail information on the interfaces.

2 Overall description

The SYS_INIT component contains a register in which a message for the interrupt subroutine can be stored for example to start the initialization or save the storage (seq.) part in a non volatile memory.

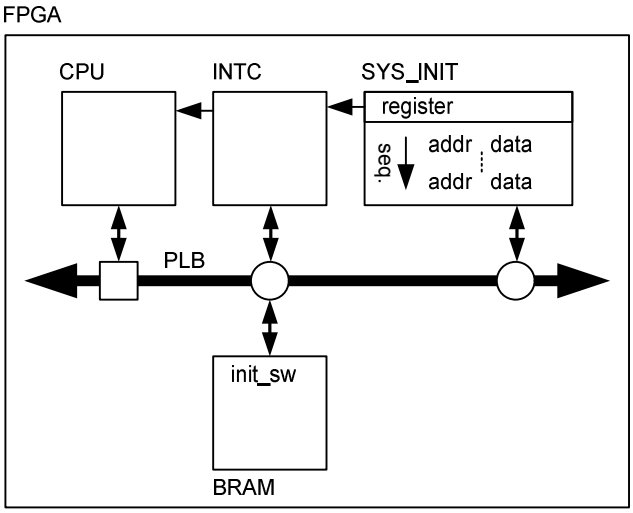

Figure 1: Board Overview

The SYS_INIT component stores not only the data (which could be a particular system setting, a parameter, a set value for the calculations, a switching command, an index for a function to call, and so on) but as well the address sequence in which the data was written and in which the processing has to happen. This is very useful in cases where the sequence is important like a feedback where first the parameters have to be set and then the function shall be called.
Sometimes a write request to the memory shall not trigger an interrupt. Hence the SYS_INIT has memory regions not resulting in a storage in the address FIFO and hence an interrupt request.

3 Request composition

The PLB memory address, to which a user writes and reads, forms a simple BRAM request. The behavior is like any other memory access. Data can be stored at a particular address and read back if desired. However, when a write to the BRAM is made with address bit A12 = ‘0’ then at the same time the address is stored in a FIFO and the interrupt line is triggered and the interrupt sink (e.g. the processor) is informed that a location in the SYS_INIT BRAM has a changed value which requires handling. The processor reads now the FIFO to find out which memory location requests the handling and due to the address does whatever is necessary (executes a function, scales the value and sends it to further devices, combines the values into a register access of a silicon chip, and so on). The FIFO makes sure the processor handles the requests according to the intended sequence.

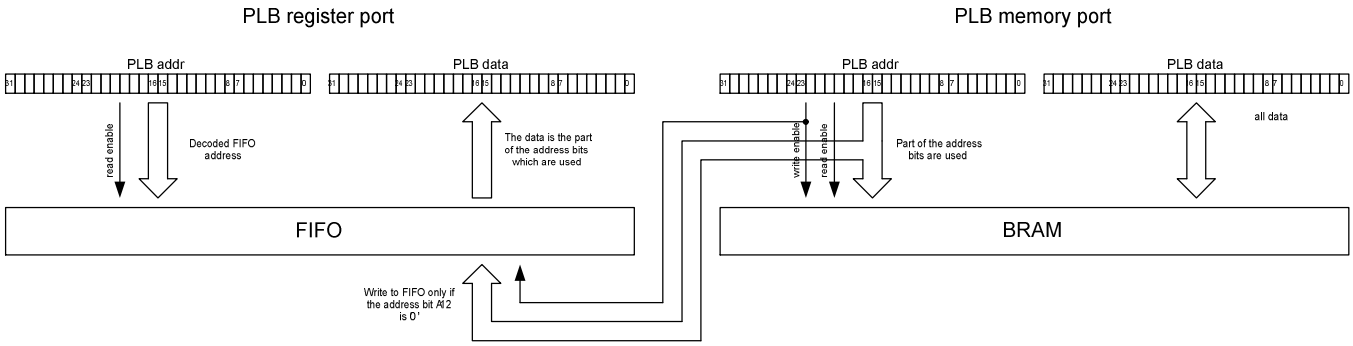

Figure 2: PLB Address Decomposition

Because some requests shall not trigger an interrupt any write with address A12 = ‘1’ will be not stored in the FIFO but in the storage location at address A11 downto A0 in the BRAM.

4 Appendix

None.