CME AHB2APB Bridge
Design Example

User Guide

11/2013
Capital Microelectronics, Inc.
China
Contents

Contents ................................................................................................................................. 2

1 Introduction....................................................................................................................... 3

2 System Level Structure .................................................................................................... 4

3 Example Result .................................................................................................................. 9

4 Pin and Design Source description .................................................................................... 10

  4.1 Pin descriptions .......................................................................................................... 10

  4.2 Pin assignments ............................................................................................................ 10

  4.3 Design source .............................................................................................................. 10

5 Revision History .................................................................................................................. 12
1 Introduction

This document describes an example that uses the AHB2APB Bridge to access the EMB on FPGA. Following is the detail of the example:

- **Function**
  - In the example, the AHB2APB Bridge module works as slave of AHB Bus and master of APB Bus, the memories on FPGA work as slave of AHB2APB Bridge.
  - ARM writes to the EMB. In the case, EMB is the APB slave and the AHB2APB Bridge has 2 slaves;
  - ARM accesses the corresponding address and reads the data from the EMB;
  - DMA transfer data from SRAM/AHB2APB Bridge slave1 to AHB2APB Bridge slave1/SRAM.
  - Compare the read data and the write data and display the results on the PC through UART.

- **The example works at**
  - FPGA Array logic: 50MHz
  - ARM core: 200MHz

- **Device: CME-M7**
2 System Level Structure

The general structure of this example as follows:

![System Level Structure Diagram]

This case consists of 5 parts as shown in above figure: PLL, SOC/ARM, APB slaves, Mem(256x32) and AHB2APB Bridge.

1. PLL (generated by Wizard):
   a) This part has an input of 20MHz and provides the clock 200MHz and 50MHz, the former is used by ARM and the latter is used in FPGA logic.

   The block can be configured as the figure 2-2 below in the Primace software by Wizard tool. In the setting parameters, PLL location is set "2".
2. SOC/ARM (generated by Wizard):
   a) This part is an ARM M3 processor core, which contains some hard peripherals. In the example, a UART2 peripheral is used and FPGA logic connects with the AHB0 slave interface.
   b) It is configured as the figure 2-3(a), 2-3(b), 2-3(c).
c) The c code in the firmware file “main.c” is used to achieve the function of accessing the EMBs on the FPGA. The function flow as figure 2-4.
3. APB slaves (RTL design)
   a) This module is designed by RTL in FPGA logic, which gives all slaves’ select and other control signals. It instantiates EMBs as APB slaves.

4. Mem (256x32, sp, generated by Wizard):
   a) This block is a 256 x 32bit memory used as AHB2APB Bridge Slave memory space, which is generated by Primace Wizard tool, and the figure 2-5(a) show the detail configuration of this module in Primace EMB Wizard.
5. AHB2APB Bridge (generated by Wizard)
   a) This module transforms the AHB interface signals to APB interface signals. The former are from the ARM SOC and the later connect with the memory port.
   b) It works as AHB slave and APB master.
   c) It works with 2 APB slaves which are both EMB.
   d) It can be configured as figure 2-6 below.
3 Example Result

The results can be displayed on the PC through serial port, using the Tera Term tool which can be configured as figure 3-1 below. The results are shown in the figure 3-2.

![Figure 3-1 Tera Term configuration](image)

```
/****************************/
| FP0-slave single and burst test begin |
/****************************/
Init done !!!

Test external FP0 slave space !!!
Finish writing (single) external FP0 space (AHB2APB Bridge slave0) !!!
Finish writing (single) external FP0 space (AHB2APB Bridge slave1) !!!
Finish reading (single) external FP0 space (AHB2APB Bridge slave0) and comparison is ok !!!
Finish reading (single) external FP0 space (AHB2APB Bridge slave1) and comparison is ok !!!

Test BRAM writing and reading !!!
Finish writing, then reading and check...
BRAM TEST PASSED !!!

Test moving data from BRAM to external FP0-slave (AHB2APB Bridge slave1) through DMA !!!
Finish initiating memory !!!
DMA Transfer Begin !!!
DMA Transfer Finish !!!
DMA data verification is OK.

Test moving data from external FP0-slave (AHB2APB Bridge slave1) to BRAM through DMA !!!
Finish initiating memory !!!
DMA Transfer Begin !!!
DMA Transfer Finish !!!
DMA data verification is OK.

END. All functions are OK.
/****************************/
```

![Figure 3-2 Example results](image)
4 Pin and Design Source description

4.1 Pin descriptions

Table 4-1 The AHB2APB Bridge example top module pin description

<table>
<thead>
<tr>
<th>Name</th>
<th>Direction</th>
<th>Width</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>clk</td>
<td>Input</td>
<td>1</td>
<td>Clock input (20MHz)</td>
</tr>
<tr>
<td>rst_n</td>
<td>Input</td>
<td>1</td>
<td>Reset signal, low active</td>
</tr>
<tr>
<td>led</td>
<td>Output</td>
<td>2</td>
<td>Slave0 and Slave1 select signal</td>
</tr>
</tbody>
</table>

4.2 Pin assignments

The following figure 4-1 shows the detail pin assignments in IO Editor of Primace

![IO pin assignment](image)

4.3 Design source

The AHB2APB Bridge example RTL source files are shown in table 4-2.

Table 4-2 The AHB2APB Bridge example's source files description

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTL</td>
<td>./design/cme_ip_AHB2APB_bridge_v1_2slaves</td>
</tr>
</tbody>
</table>
The top module, implements the connection of all sub modules.

This module transforms AHB slave interface signals into APB interface signals.

The apb slave module, provides apb slave signals connect with AHB2APB Bridge.

The ARM processor core implemented by ARM Wizard

Single port memory(256x32) implemented by EMB Wizard

Phase-locked loop, implemented by PLL Wizard

UART initiate, and access the memory(work as APB slaves) on FPGA.
5 Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2013-11-18</td>
<td>Initial release</td>
</tr>
</tbody>
</table>
