

MSE - TSM_EmbHardw

LCD interface design

TABLE OF CONTENTS

1 Using the Controller Core

2

Introduction

This document gives some advice to design a controller core to communicate with the LCD display. The LCD display itself has a controller to control the pixels etc. This fact is going to simplify our controller implementation as well as a real-world example. Therefore our task is going to be configuring the slave controller. Afterwards we will send the image data to the LCD for displaying it.

- ▶ Propose a register model (bus Avalon) and a schematic. **ON A PAPER!**
 - ▶ Clearly identify the inputs and outputs on the interface.
 - ▶ Include an output *waitrequest* on the Avalon interface.
 - ▶ The Avalon bus must not perform a new access until the writing of former data beat hasn't been completed
- ▶ Write the VHDL code for Avalon interface and the logic required for generating the correct timing of the LCD interface signals
- ▶ Simulate with ModelSim and verify the time-diagram.
- ▶ Add to Qsys and re-generate the SoPC.

1 Using the Controller Core

- ▶ Initialize the LCD:
 - ▶ Refer to the initialisation exemple on Moodle
- ▶ On eclipse: send some pixels. For doing so use the command allowing to write to the graphic memory followed by two intricate loops for swapping every possible value for x and y .
- ▶ A real image cannot be generated as a pattern. It must be stored somewhere in memory as an array and the processor must first read it from memory to write it on the LCD.
- ▶ Send some pixels from the data memory. For doing so, initialize a two-dimensional array with some values, and then send the vector values to the LCD interface.

If after a correct simulation the LCD doesn't work, debug the interface with Eclipse and the SignalTap Analyzer:
Verify that the behavior on the real circuit matches the simulated behavior.