Embedded Systems
Processor Architecture Laboratory

DE0-Nano-SoC Lab3 - Design of master unit interface for FPGA

Teacher :

Beuchat René

Student :

Lefebure Nicolas
Lamya Omari

8 janvier 2020

Table des matières

1 Introduction

The goal of this lab is the design of custom master interface. Unlike the slave interface, it is targeted towards complex use cases where large amounts of data need to be moved. An LCD display controller and a camera controller associated with a Nios II processor need to be design. In this report in particular we focus on the display part (LCD), which is the displacement of the data from a memory to a peripheral.

2 Overall operation and elements

The implementation of the LCD controller is done with the following components :

- An LT24 LCD module : It is provided with a ILI9341 LCD driver and has a resolution going up to 320x240.
- DE0-Nano-Soc board.

To resume basically the overall design, the images are firstly obtained by the camera module. Our DMA unit will allow the controller to read the data from the Memory (SDRAM) by using the Avalon Bus. Then it will communicate with the LCD driver by respecting the 8080 I protocol.

3 Full system block diagram

The full system block diagram is shown in the figure 1. The camera sends the pictures to the Memory of the FPGA. The data is then accessible by the LCD that will display the pictures.

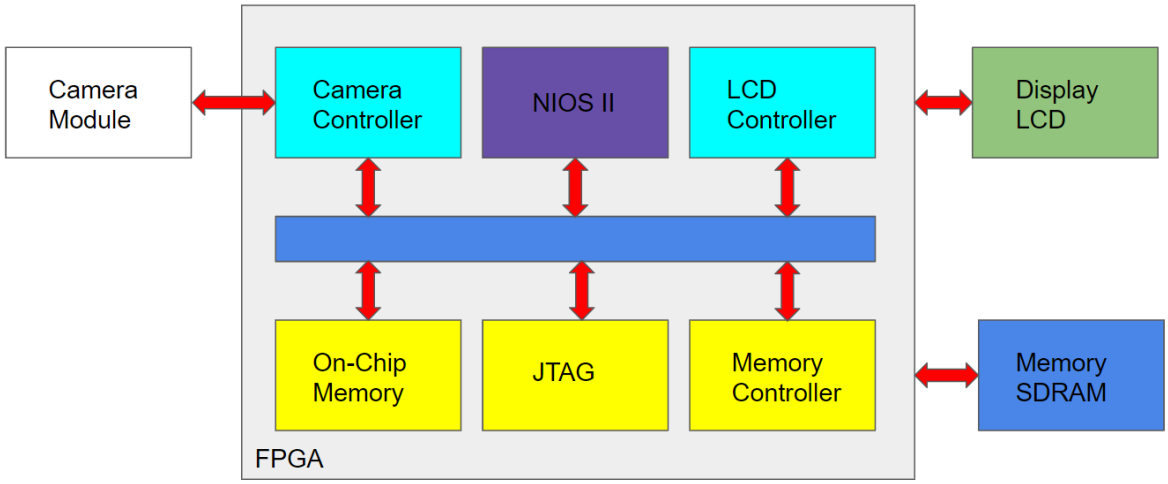

FIGURE 1: Full system block diagram

4 Custom IP

4.1 Custom IP block diagram

The custom IP component will be made of a slave and a master Avalon interface. The master will be used to communicate the data through the Avalon Bus while the slave actually has two functionality : writing and reading configuration registers from the DMA and writes set up configurations in the LT24 module. The block diagram of the component is visible in the figure 2.

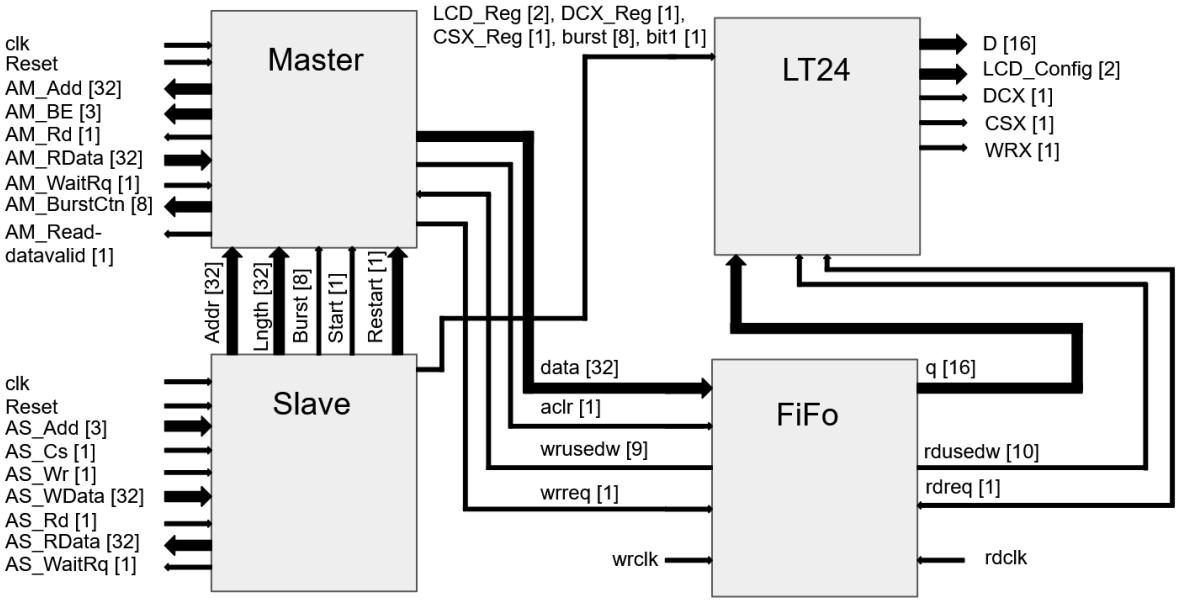

FIGURE 2: Custom IP block diagram

4.2 Custom IP register map

The table below shows the register map of the IP component. The first five registers correspond to configurations for the DMA while the following registers correspond to configurations for the LT24 module.

Name	Offset interface	Offset uP	Mode	Data width	Description
Image address	0	0	R/W	32	Memory address indicating where to start to put the data (image).
Length	1	4	R/W	32	Length of the memory in order to to stock the data.
Burst	2	8	R/W	8	The data are sent packet by packet by burst of pixels continuously.
Start	3	12	R/W	1	Bit that indicates when to start the transfer of data from memory to LCD.
Restart	3	12	R/W	1	Bit that is used to restart DMA when new configuration is needed.
LCD_Reg	4	16	W	2	To control LCD_On and LCD_Reset
DCX_Reg	4	16	W	1	Bit that should be 0 when we send a command and 1 when we send data.
CSX_Reg	4	16	W	1	To control chip select during different command
burst	4	16	W	8	The data are sent packet by packet by burst of pixels continuously.
Bit 1	4	16	W	1	Bit to differentiate between writing to LCD_Reg and to other registers.

4.3 Custom IP finite state machine (FSM) diagram

4.3.1 DMA state machine

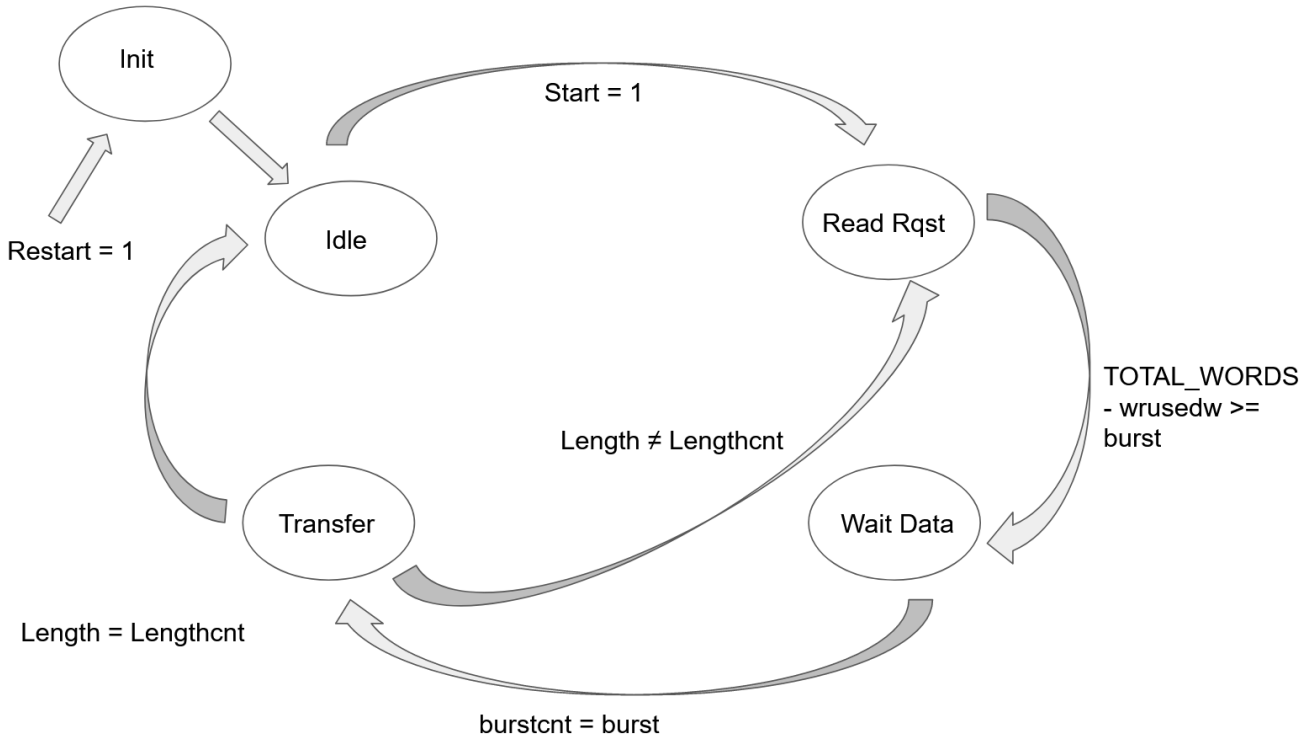

FIGURE 3: Final state machine of DMA

— Idle : The state machine starts with the Idle state (or with the state Init if new configurations need to be set up). In this state, all the counters are set to zeros (burst counter and length counter) and the starting address where the data is taken is defined. When the 'start' bit is enable, the data is ready to be transferred and the state goes to 'Read Request'.
— Read Request : Here the controller send a read request to the Avalon Bus by triggering the 'AM_Rd' bit. The Avalon master is also given the starting address and the burst length. Once there is enough space in the FiFo, the state goes to 'Wait Data'.
— Wait Data : Here, the data coming from the Avalon is written into the FiFo. When the burst counter has attained the total burst length, the state goes to 'Transfer'.
— Transfer : In this state, the total image length counter is incremented along with the address. Then a test is effectuated : if the total length of the image is reahc the state goes to 'Idle' however if it is not the case the state goes back to 'Read Request', in order to get another burst.
— Init : This state is use to restart DMA when a new configuration is needed. It clears the FiFo by enabling signal 'aclr'. It automatically goes to 'Idle' state.

4.3.2 LT24 functionality

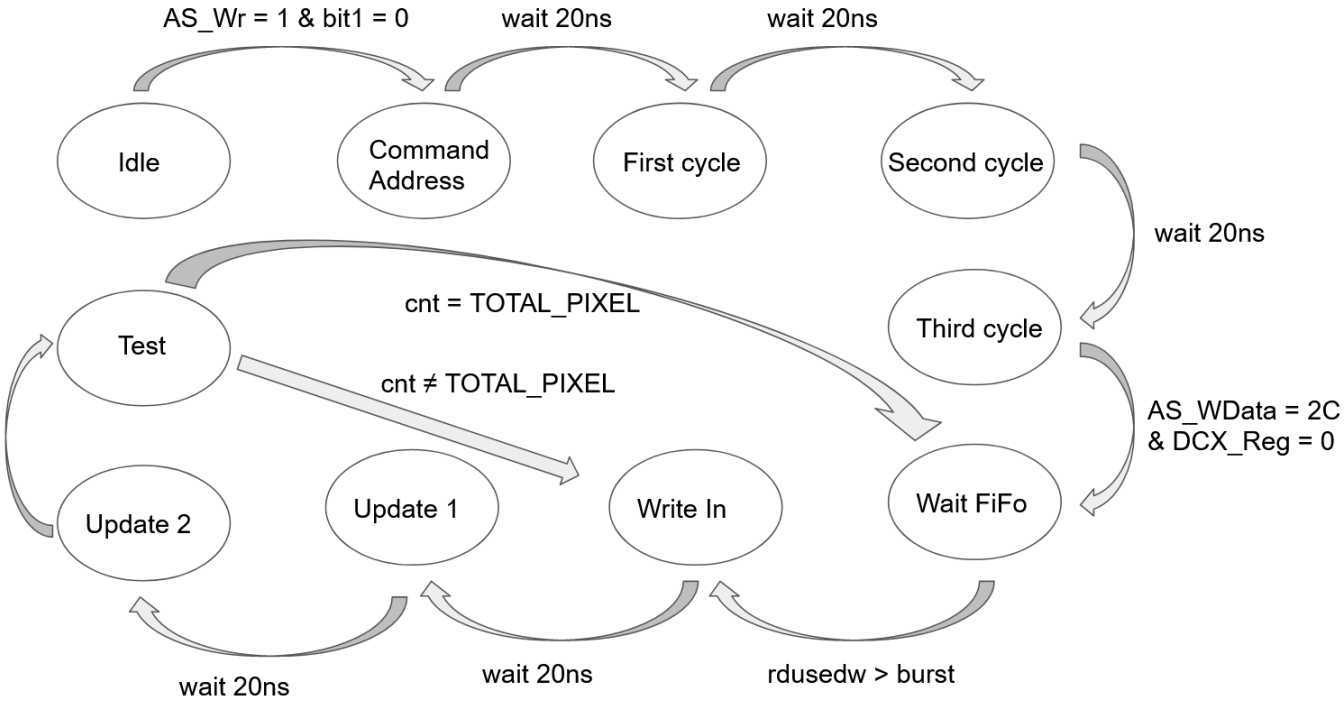

FIGURE 4: Final state machine of LT24

The LT24 interface will make the connection between the FiFo and the LCD. It receives specifications from the slave interface.

- Idle : The state machine starts with the 'Idle' state. In this state, the pixel counter 'cnt' is set to zero. The protocol starts with every signal to 1 (CSX, DCX, WRX). When the 'AS_Wr' bit is triggered, the module is reay to receive instructions and the state goes to 'Command Address'.
- Command Address : In this state a command address is expected, so 'CSX', 'WRX' and 'DCX' signal are put to zero.
- Three cycles : The controller then waits four cycles of 80 ns before going to 'Wait FiFO' state, in order to respect the timing protocol. The controller enters in the 'Wait FiFo' state only if the command is 2Ch : Memory write. Otherwise it goes back to 'Idle' for other commands.
- Wait FiFo : In this state, 'CSX' signal is put to 1 to pause parallel interface. When there is more than a burst of information in the FiFo, the sate goes to 'Write In'.
- Write In : Here a command data is expected so the signals 'CSX' and 'WRX' are put to zero while 'DCX' is put to 1. The transfer is made from the FiFo to the memory of the LT24 module.
- Two updates : The controller then waits four cycles of 80 ns before going to 'Wait FiFO' state, in order to respect the timing protocol.
- Test : In this state, the total pixel counter is incremented. Then a test is effectuated : if the total pixel is reach the state goes to 'Wait FiFo' however if it is not the case the state goes back to 'Write In'.

4.3.3 Information about FIFO

We will use a dual clock FIFO when implementing our design. Quartus actually provides a FIFO generator which we will directly use for this purpose. It will have an input data width of 32 bits while an output data width of 16 bits, as the Camera group send us 32 bits and the LT24 input can receive up to 16 bits. The word size has to be larger than a burst. It was set to 512 words to ensure un good marging. To get information about the filling of the FiFo, two signal are used : wrusedw and rdusedw, which directly indicate the number of words inside the FiFo. A last signal is used to clear the FiFo (aclr) when a 'restart' is effectuated to the system.

5 Data format

The 65K-Color, RGB 5-6-5 bits input data based on 8080-system 16-bit parallel bus interface of ILI9341 will be used on the LT24. As the LCD device requires a 16 bit wide RGB data [5,6,5] input (register D[15...0]), we will pull 16 bits from the dual FiFo unit.

Count	0	1	2	3	...	238	239	240
D/CX	0	1	1	1	...	1	1	1
D15		0R4	1R4	2R4	...	237R4	238R4	239R4
D14		0R3	1R3	2R3	...	237R3	238R3	239R3
D13		0R2	1R2	2R2	...	237R2	238R2	239R2
D12		0R1	1R1	2R1	...	237R1	238R1	239R1
D11		0R0	1R0	2R0	...	237R0	238R0	239R0
D10		0G5	1G5	2G5	...	237G5	238G5	239G5
D9		0G4	1G4	2G4	...	237G4	238G4	239G4
D8		0G3	1G3	2G3	...	237G3	238G3	239G3
D7	C7	0G2	1G2	2G2	...	237G2	238G2	239G2
D6	C6	0G1	1G1	2G1	...	237G1	238G1	239G1
D5	C5	0G0	1G0	2G0	...	237G0	238G0	239G0
D4	C4	0B4	1B4	2B4	...	237B4	238B4	239B4
D3	C3	0B3	1B3	2B3	...	237B3	238B3	239B3
D2	C2	0B2	1B2	2B2	...	237B2	238B2	239B2
D1	C1	0B1	1B1	2B1	...	237B1	238B1	239B1
D0	C0	0B0	1B0	2B0	...	237B0	238B0	239B0

FIGURE 5: - Data format conversion from the SDRAM to the LT24

6 LCD and camera synchronisation

Every time that a new address will be written into the Image address register, the start Bit is set to 1 and the DMA unit then starts taking the data from Memory. In that situation, two possible cases can occur, as shown in the figure 6.

The figure 6 shows the two situations.

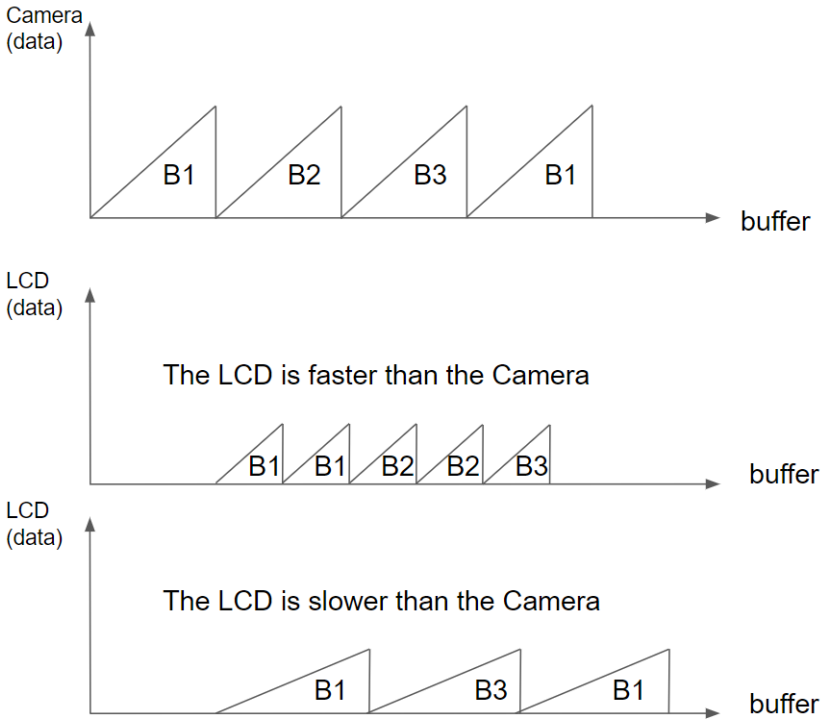

FIGURE 6: Camera and LCD data communication

7 Top-level connections

The figure 7 shows the connection between the LT24 device and the DE0-Nano board.

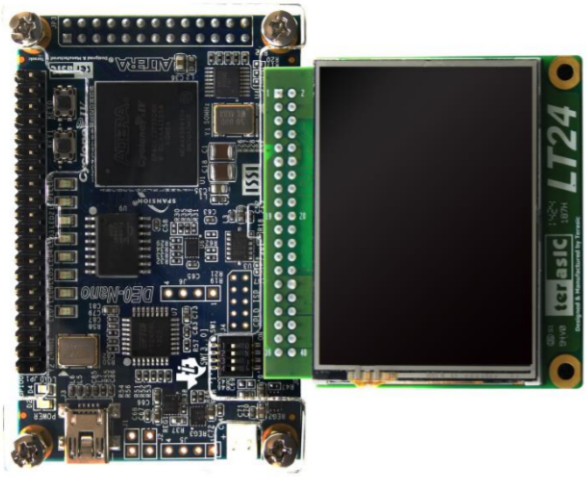

FIGURE 7: Connection between LT24 and DE0-Nano

8 Conclusion

This report summarizes a good part of the ideas that we had before embarking on the true conception of the project. We are aware that information is surely lacking. That is why we have tried to make the project as modular as possible (adding more register, signals) to avoid unpleasant surprises during the realization or when assembling everything with the camera group.