Digital System Design
SS 2023

Lab 2 : Parallel Processing (Digital Audio Filter)

Report submitted by:

Riya Alias

Matrikelnr: 1119262

Geethusagar Ajitha

Matrikelnr: 1119258

Najmeddine Bouzambila
Matrikelnr: 738345

June/19/2023

Content

1 Introduction

The main purpose of this lab is to display selected filter number on the attached 7-segment display. The 7- segment display is attached to the Pmod JA1+JB1 of ZedBoard. Parallel processing in FPGAs is demonstrated by analyzing the implementation of algorithms for Digital filters.

2 Implementation

2.1 I2S data format for audio data exchange

(a):

Task-Description(a): in this task we will explain the data format I2S for audio data exchange between Audio Codec and Zynq device and to name data signals and clock signals of this data format by studying ADAU1761 product description.

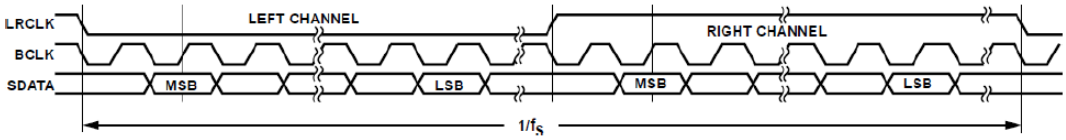

Figure 1: I2S Mode—16 Bits to 24 Bits per channel

- LRCLK: Serial Data Port Frame Clock
- BCLK: Serial Data Port Bit Clock
- SDATA: Audio serial data

Serial data port Bit clock, BCLK has a frequency configured in accordance with the desired sampling bit rate. On every Negative edge trigger of BCLK one audio data bit is transmitted. The whole frame is sent twice, one to the left channel and one to the right channel of the Zync DSP block. LRCLK has duty cycle of 50% and on every negative edge of LRCLK data is sent to right channel and on every positive edge of LRCLK data is sent to left channel.

2.2 Analysis and understanding RTL Design

Task-Description(b): in this task we will generate the RTL schematic of the instance adau1761_test.vhd and to understand the design used for audio data processing.

Elaboration: In order to generate the RTL schematicwe first open the project file audio24bit in Vivado. Under the Flow Navigator we triggered the RTL Analysis and Elaborated Design is opened.

Questions:

What is the purpose of multiplexer hphone_l_i and hphone_r_i?

Input is being sampled by the audio codec from input GPIO pins. Sampled inputs are subjected to low pass filters to cancel the external noise associated with it. There are three filters with different lengths and filtering capabilities and and one dsp block with no filtering.

Any one of the three filters or the block with no filtering can be selected by the user. Hphone_l_i and Hphone_r_i takes the filtered outputs as input and user filter choice as selection bits and delivers corresponding output.

00	No filtering
01	Filter with length 8
10	Filter with length 16
11	Filter with length 32

Table 1: User Selection MUX output

What is the function of Inst_dsp_block?

It takes AD converted signal from Audio Codec and sends the same to output without filtering.

Which port is connected to the serial data output of ADC?

ADC_SDATA/GPIO1 (pin 26): ADC Serial Output Data (ADC_SDATA).

Which port is connected to the serial data input of DAC?

DAC_SDATA/GPIO0 (pin 27): DAC Serial Input Data (DAC_SDATA).

What is the clock frequency of MCLK? What is the sample frequency of the ADC and DAC?

Clock frequency of MCLK is 24MHz and sampling frequency of ADC and DAC is 48 KHz.

What is the filter length N of the three low pass filters? Calculate the cut-off frequency of each filter.

Filter lengths N of the three low pass filters are 8, 16 and 32 respectively.
Cutoff Frequency, Fc = sampling frequency/ filter length

For Filter lengths
N =8, Fc = 6 KHz
N=16, Fc = 3 KHz
N=32, Fc = 1.5 KHz

2.3 Top Level Inputs and Outputs

Task-Description(c): Print out of the RTL schematic and each top level input and output description!

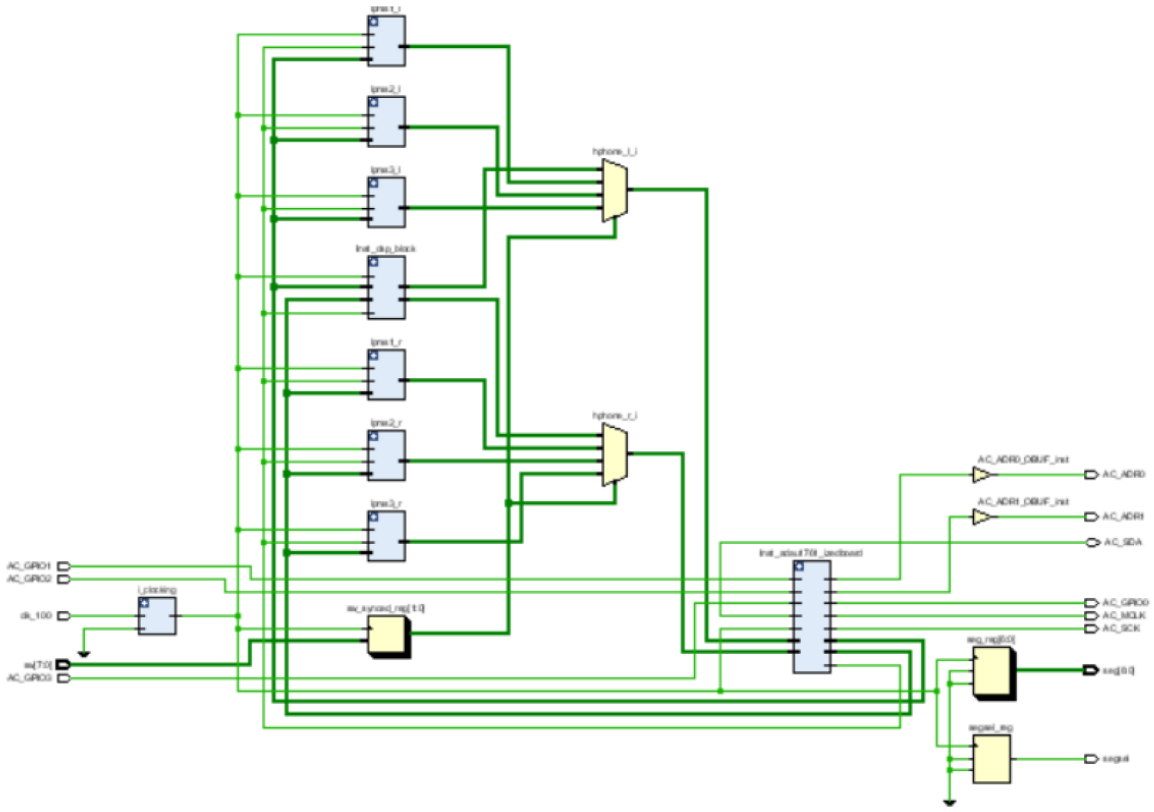

Figure 2: RTL Schematic

Input Signals	Description
AC_GPIO1	Digital Audio Serial Data ADC Output (I2S MOSI)
AC_GPIO2	Digital Audio Bit Clock Input/Output (I2S_bclk)
clk_100	Clock configuration
sw [7:0]	filter select
AC_GPIO3	Digital Audio Left-Right Clock Input/Output (I2S_LR)

Table 2: Input Signals Information

Output Signals	Description
AC_ADR0	I2C Address Bit 0/SPI Latch Signal
AC_ADR1	I2C Address Bit 1/SPI Data Input
AC_SDA	I2C Serial Data Interface(bidirectional)
AC_GPIO0	Digital Audio Serial-Data DAC Input (I2S MISO)
AC_MCLK	Master Clock input(codec)
AC_SCK	I2C Serial Data interface
seg [6:0]	7 segments
segsel	cathode

Table 3: Output Signals Information

2.4 Interfacing Seven Segment Display

Task-Description(d): To display selected filter number on the 7-segment display, this is attached to the Pmod JA1+JB1 of ZedBoard and to add IO port settings for communication with pmod.

Elaboration: Connection diagram of seven segment display and IO configuration for pmod JTAG communication are as illustrated below.

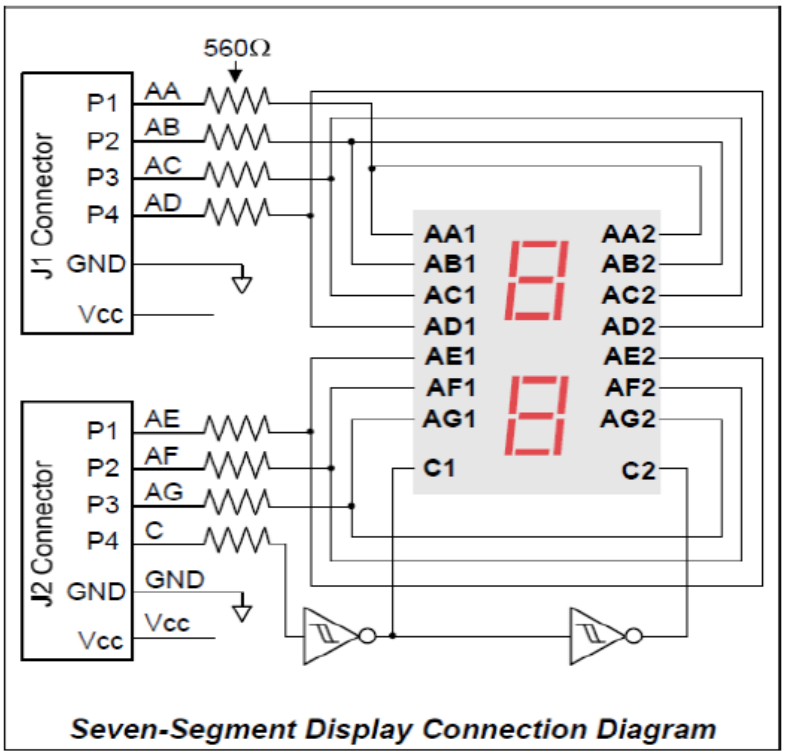

Figure 3: Connection of the Seven Segment Display

Name	Direction	Site	I/O Standard
Seg[0]	output	Y11	LVCMOS33
Seg[1]	output	AA11	LVCMOS33
Seg[2]	output	Y10	LVCMOS33
Seg[3]	output	AA9	LVCMOS33
Seg[4]	output	W12	LVCMOS33
Seg[5]	output	W11	LVCMOS33
Seg[6]	output	V10	LVCMOS33

Table 4: I/O ports Configuration

```
begin
process(clk_48)
   begin
      if rising_edge(clk_48) then
         sw_synced <= sw;
            ------------------make your changes here
            segsel<='0'; -- define the segment (cathode)
            case sw is
            WHEN "00000000"=>seg<="0111111"; -- control 7-Segment
            WHEN "00000001"=>seg<="0000110";
            WHEN "00000010"=>seg<="1011011";
            WHEN "00000011"=>seg<="1001111";
            WHEN OTHERS=>seg<="1111111";
            END CASE;

         --------------------------------
      end if;
   end process;
```

Figure 4: Seven Segment Display code snippet

2.5 Synthesis and Analysis of Utilization report

Task-Description(f): To synthesis the design and to analyze the utilization report and explain about Slice LUTs, Slice Register, Block RAM and DSPs.

Elaboration:

1.Slice LUT's: LUT's defined as Look Up Tables store predefined outputs for every combination of inputs. LUT in FPGA is a collection of hard-wired logic gates. Slice LUT is a collection of such LUT's.

2.Slice Registers: Slice Registers are group of flip-flops which store the bit pattern. For every clock cycle the input data is latched and is stored internally and the output data is updated to match the internally stored data. Utilization of Slice LUT's and Slice Registers is shown in the picture below.

```
1. Slice Logic
--------------

+----------------------------+------+-------+-----------+-------+
|          Site Type         | Used | Fixed | Available | Util% |
+----------------------------+------+-------+-----------+-------+
| Slice LUTs*                |  677 |     0 |     53200 |  1.27 |
|   LUT as Logic             |  528 |     0 |     53200 |  0.99 |
|   LUT as Memory            |  149 |     0 |     17400 |  0.86 |
|     LUT as Distributed RAM |    0 |     0 |           |       |
|     LUT as Shift Register  |  149 |     0 |           |       |
| Slice Registers            |  547 |     0 |    106400 |  0.51 |
|   Register as Flip Flop    |  547 |     0 |    106400 |  0.51 |
|   Register as Latch        |    0 |     0 |    106400 |  0.00 |
| F7 Muxes                   |    0 |     0 |     26600 |  0.00 |
| F8 Muxes                   |    0 |     0 |     13300 |  0.00 |
+----------------------------+------+-------+-----------+-------+
```

Figure 5: Utilization of Slice LUT's and Slice Registers

3.Block RAM: FPGA contains two types of internal RAM's. Block RAM and Distributed RAM. Block RAM serves as a relatively larger memory structure. They are used to store a bunch of data on chip for example synthesizing memory and FIFO functions. RAM utilization is shown in the picture below.

```
2. Memory
---------

+-------------------+------+-------+-----------+-------+
|     Site Type     | Used | Fixed | Available | Util% |
+-------------------+------+-------+-----------+-------+
| Block RAM Tile    |  0.5 |     0 |       140 |  0.36 |
|   RAMB36/FIFO*    |    0 |     0 |       140 |  0.00 |
|   RAMB18          |    1 |     0 |       280 |  0.36 |
|     RAMB18E1 only |    1 |       |           |       |
+-------------------+------+-------+-----------+-------+
```

Figure 6: Utilization of BLOCK RAM

4.DSP: These blocks are used for DSP applications; they can be used to implement various complex calculations at a faster rate. They are highly optimized for maximum performance and minimum resource utilization. DSP Block utilization is shown below.

```
3. DSP
------

+-----------+------+-------+-----------+-------+
| Site Type | Used | Fixed | Available | Util% |
+-----------+------+-------+-----------+-------+
| DSPs      |    0 |     0 |       220 |  0.00 |
+-----------+------+-------+-----------+-------+
```

Figure 7: Utilization of DSP

1.1 Summary of Registers by Type

Total	Clock Enable	Synchronous	Asynchronous
0	_	-	-
0	_	-	Set
0	_	-	Reset
0	_	Set	-
0	_	Reset	-
0	Yes	-	-
0	Yes	-	Set
0	Yes	-	Reset
10	Yes	Set	-
537	Yes	Reset	-

2. Memory

Site Type	Used	Fixed	Available	Util%
Block RAM Tile	0.5	0	140	0.36
RAMB36/FIFO*	0	0	140	0.00
RAMB18	1	0	280	0.36
RAMB18E1 only	1			

5. Clocking

Site Type	Used	Fixed	Available	Util%
BUFGCTRL	2	0	32	6.25
BUFIO	0	0	16	0.00
MMCME2_ADV	1	0	4	25.00
PLLE2_ADV	0	0	4	0.00
BUFMRCE	0	0	8	0.00
BUFHCE	0	0	72	0.00
BUFR	0	0	16	0.00

6. Specific Feature

Site Type	Used	Fixed	Available	Util%
BSCANE2	0	0	4	0.00
CAPTUREE2	0	0	1	0.00
DNA_PORT	0	0	1	0.00
EFUSE_USR	0	0	1	0.00
FRAME_ECCE2	0	0	1	0.00
ICAPE2	0	0	2	0.00
STARTUPE2	0	0	1	0.00
XADC	0	0	1	0.00

Task-Description(h): Program Target Device and Take photo of 7-segment display.