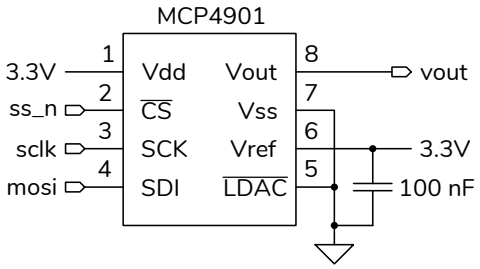
SPI Interface

Introduction

In this lab you will design and implement a transmit-only SPI interface. You will use it to set the output voltage of an MCP4901 8-bit digital-to-analog converter (DAC).

The MCP4901 has the following pinout:



- V_{dd} and V_{ss} are the digital supply (3.3 V) and ground voltages respectively
- V_{ref} is the maximum analog output level. This will be connected to the 3.3 V supply.
- V_{out} is the analog voltage output whose value is $V_{ref} \times {}^{d}/_{256}$ where d is the 8-bit digital value written to the DAC.
- $\overline{CS}$, SCK and SDI, correspond to the $\overline{SS}$, SCLK and MOSI SPI interface signals
- $\overline{LDAC}$ should be set low

The value written to the MPC4901 must be a 16-bit value constructed as defined as in the diagram below taken from its Datasheet:

REGISTER 5-3: WRITE COMMAND REGISTER FOR MCP4901 (8-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
0	BUF	$\overline{GA}$	$\overline{SHDN}$	D7	D6	D5	D4	D3	D2	D1	D0	x	x	x	x
bit 15															bit 0

The most significant four bits should be set to `4'b0011`. The least-significant four bits are "don't care" (set them to `4'b0000`).

In this lab you will connect the DAC's SPI interface to your CPLD, implement an SPI interface on the CPLD and use it to set the DAC's output voltage to a value determined by the digits of your BCIT ID. You will measure the analog voltage output with a DMM to verify the correct operation of the interface.

You will be supplied with a `lab7.sv` file that defines an incomplete `spi` module, a `lab7` top-level module, a `clkdiv` clock divider to generate a 1 MHz clock, and a `lab7_tb` testbench. You need only add the code that implements an `spi` module which is the datapath shown below[1]:

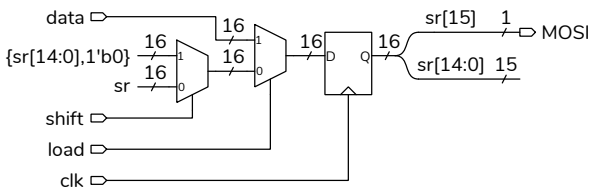


Figure 1 shows simulation results when writing the value 16'h3B20.

Requirements

Pushing keypad key [1] should result in an zero voltage output from the DAC. Pushing keypad key [2] should result in a voltage output equal to the last two digits of your student ID modulo 32 (the remainder after dividing by 32) plus one divided by 10. For example if your student ID were A00123456 then (56 mod 32) + 1 = 25 and the output voltage should be 2.5 V.

The analog output voltage is given by the equation:

$$V_{out} = V_{ref} \frac{d}{256}$$

where d is the 8-bit integer written to the DAC in bits 11 through 4 of the 16-bit word. For example, to obtain a 2.5 V output the value d would be:

$$d = 256 \frac{V_{out}}{V_{ref}} = 256 \frac{2.5}{3.3} = 194 = 8'b1100\_0010$$

and the 16-bit word written to the DAC would be 16'b0011_1100_0010_0000.

[1] This is a simplified version of the one in the Interfaces lecture notes.

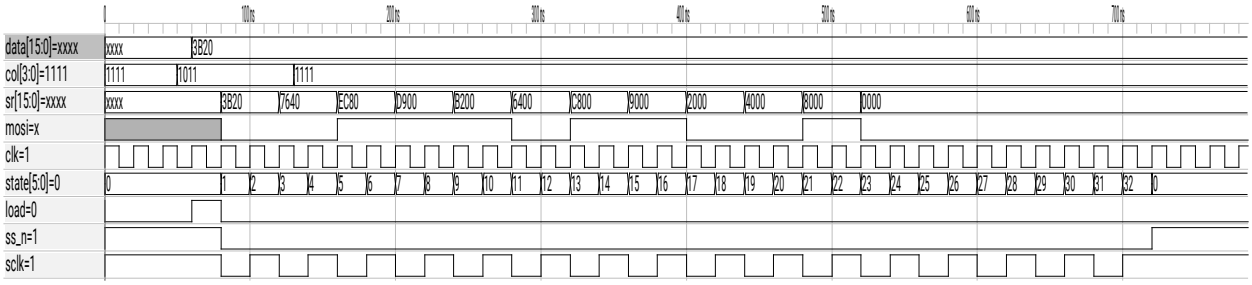
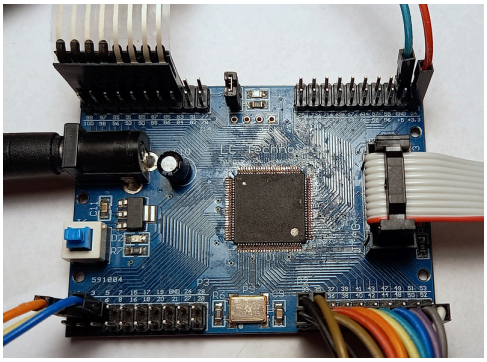
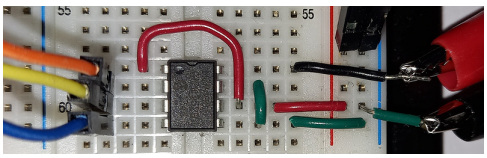


Figure 1: Simulation Results.

CPLD I/O

The following photos shows the connections between the CPLD board and prototyping board.





The **row**, **col** and **clk50** pin connections are the same as in previous labs. The following additional pin assignments are suggested:

CPLD Pin	MPC4901 Pin	Signal Name
1	2	**ss_n**
3	3	**sclk**
5	4	**mosi**

A **lab7.qsf** file is available on the course web site with these pin assignments.

The ground and 3.3 V connections can be made to the pins at the top right of the CPLD board. The 100 nF bypass capacitor will reduce noise on V_{dd} and V_{ref}. Do *not* use external power supplies.

Procedure

Download the **lab7-incomplete.sv** file and save it as **lab7.sv**. Create a Quartus project named **lab7** and add the file **lab7.sv**. You can import the pin assignments in the **lab7.qsf** file.

Add code to the **spi** module to implement the block diagram shown above. Edit the line **data <= 16'b....** and insert the value corresponding to your BCIT ID.

Wire up the MCP4901 DAC from your ELEX 2117 parts kit as shown above and connect power, ground, and the $\overline{\text{CS}}$, SCK and SDI signals to the appropriate CPLD board pins.

Connect a DMM to V_{out} to measure the output voltage. Pressing the 1 key should result in the DMM displaying 0 V. Pressing the 2 key should result in the DMM displaying the appropriate voltage for your student number.

The supplied **lab7.sv** file also contains a testbench (**lab7_tb**) that you can use to troubleshoot your design with the test vector file (**lab7tv.csv**) on the course web site.

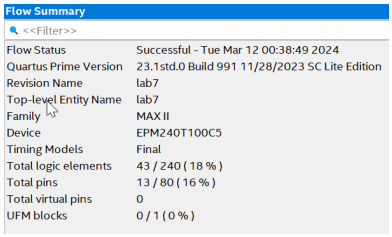
Submission

To get credit for completing this lab, submit the following to the appropriate Assignment folder on the course website:

- A PDF document containing:

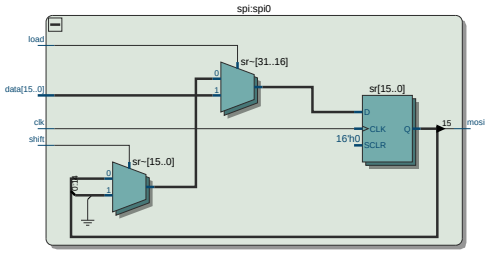
    1. The calculation of the required output voltage corresponding to your BCIT ID and the 16-bit value that you need to write to the DAC, computed as described above.

2. A listing of your Verilog code for the **`spi`** module (include only the **`spi`** module, not the other code supplied).
3. A screen capture of your compilation report (Window > Compilation Report) similar to:



Flow Summary	
<<Filter>>	
Flow Status	Successful - Tue Mar 12 00:38:49 2024
Quartus Prime Version	23.1std.0 Build 991 11/28/2023 SC Lite Edition
Revision Name	lab7
Top-level Entity Name	lab7
Family	MAX II
Device	EPM240T100C5
Timing Models	Final
Total logic elements	43 / 240 (18 %)
Total pins	13 / 80 (16 %)
Total virtual pins	0
UFM blocks	0 / 1 (0 %)

4. A screen capture of the schematic created by Tools > Netlist Viewers > RTL Viewer, showing only the **`spi`** module (use the + button). For example:



- If you do not demonstrate your completed lab in person, submit a short video showing the keypad and DMM voltage display as you press the 1 and 2 keys.