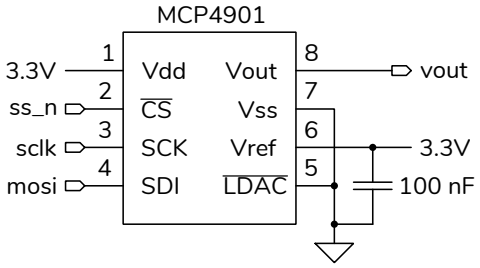
SPI Interface

Version 2: Corrected value written to DAC for 0V.

Introduction

In this lab you will design and implement a transmit-only SPI interface. You will use it to set the output voltage of a Microchip MCP4901 8-bit digital-to-analog converter (DAC). This is an binary-weighted DAC with an internal 8-bit register that is loaded over an SPI interface.

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 (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=4'h3`. The least-significant four bits are "don't care" (set them to `4'b0000=4'h0`).

In this lab you will connect your CPLD to the DAC's SPI interface, implement an SPI interface on the CPLD and use it to set the DAC's output voltage to a value determined by the last two 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 Quartus project archive, `lab7.qar`, that includes an incomplete `spi` module in the file `spi.sv`, a `lab7` top-level module, and a `clkdiv` clock divider to generate a 1 MHz clock. On the web site you'll also find a `lab7_tb` testbench so that you can simulate your design if necessary. You need only add the code that implements an `spi` module which is the datapath shown below[1]:

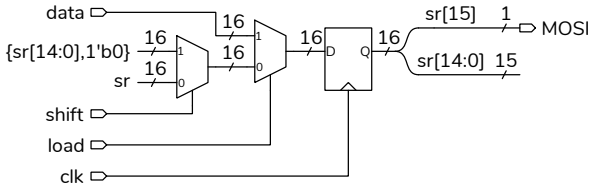


Figure 1 shows simulation results when writing the value 16'h3C20 to the DAC over the SPI interface.

Requirements

The `lab7` module writes the DAC register with 16'h3000 when [1] is pushed and this results in a 0 V output from the DAC.

The module writes the value `MYV` to the DAC when keypad key [2] is pushed. You must set this value of (near the top of `lab7.sv`) so that pushing [2] results in a voltage output equal to:

$$V = \frac{(i \bmod 32) + 1}{10} \text{ Volts}$$

where i is the last two digits of your student ID and mod is the modulo operation. For example if your

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

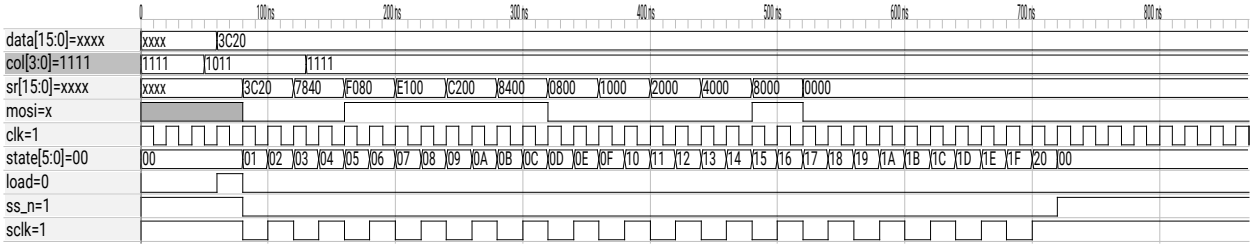


Figure 1: Simulation Results.

student ID were A00123456 then the output voltage should be 2.5 V ($\frac{(56 \mod 32)+1}{10} = 2.5$).

The analog output voltage for an 8-bit binary-weighted DAC is given by the equation:

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

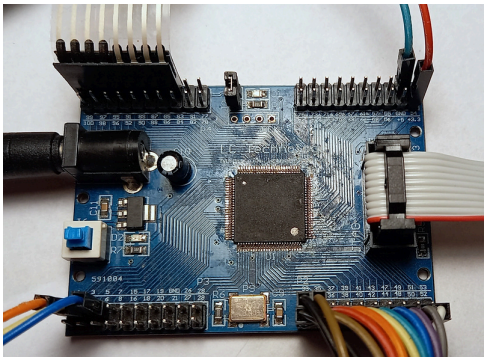
where d is the 8-bit integer written to the DAC (in bits 11 through 4 of the 16-bit word). For example, to obtain a 2.5 V output the value d would be:

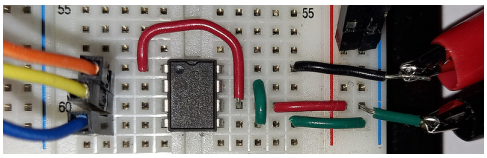
$$d = 256 \frac{V_{\texttt{out}}}{V_{\texttt{ref}}} = 256 \frac{2.5}{3.3} = 194 = 8\text{'b1100\_0010}$$

and the 16-bit word written to the DAC would be 16'b0011_1100_0010_0000=4'h3C20.

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 used in the **lab7.qsf** settings file in the project archive:

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

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 $V_{\texttt{dd}}$ and $V_{\texttt{ref}}$. Do *not* use external power supplies.

Pre-Lab

Be ready to show the following at the start of your lab:

- 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.
- A Quartus project for this lab (created when you extracted **.qar** file) including the three additional pin assignments.
- A module declaration for the **spi** module. It need not be complete or working.

Procedure

Download the **lab7.qar** file and extract the contents to a working directory.

Add code to the **spi** module to implement the block diagram shown above.

Edit the file **lab7.sv** and modify the line beginning with **localparam MYV =** near the top of the file

to set **MYV** to the 16-bit value that must be sent over the SPI interface to produce the required output voltage.

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 $V_{\mathbf{out}}$ 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 ID.

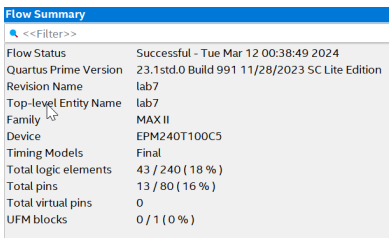
The course web site also contains a testbench (**lab7_tb.sv**) that you can use to troubleshoot your design with the test vector file (**lab7tv.csv**) on the course web site.

Submission

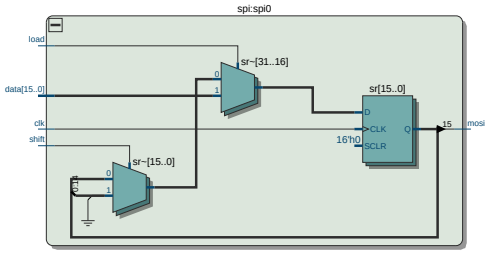
To get credit for completing this lab, submit the following to the appropriate Assignment folder on the course website:

- 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 your **spi** module, not the other code supplied).
    3. A screen capture of your compilation report (Window > Compilation Report) similar to:



    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.