On the Horizon:

- You should be watching Lecture09 videos. Quiz on Monday
- If you have not gotten checked off on Exercise16 (test of Auth_Blk on DE0-Nano) you should do so today
- You should finish today's exercise by Monday. We will map it to DE0 and test on Monday.

Exercise 17: A2D Intf Design and Test Bench (HW5 Prob3)

- We use 3 channels of the A2D converter for our Segway
  - Channel 0 → Left load cell
  - Channel 4 → Right load cell
  - Channel 5 → Battery voltage

- The A2D converter (ADC128S) on the DE0-Nano board is a SPI based device
  - We will use our SPI_mstr16 to access it
  - We will perform two transactions nearly back to back
    - First transaction we tell the ADCC128S what channel to convert
    - Second transaction we read the channel back.

This exercise can be done as a project team

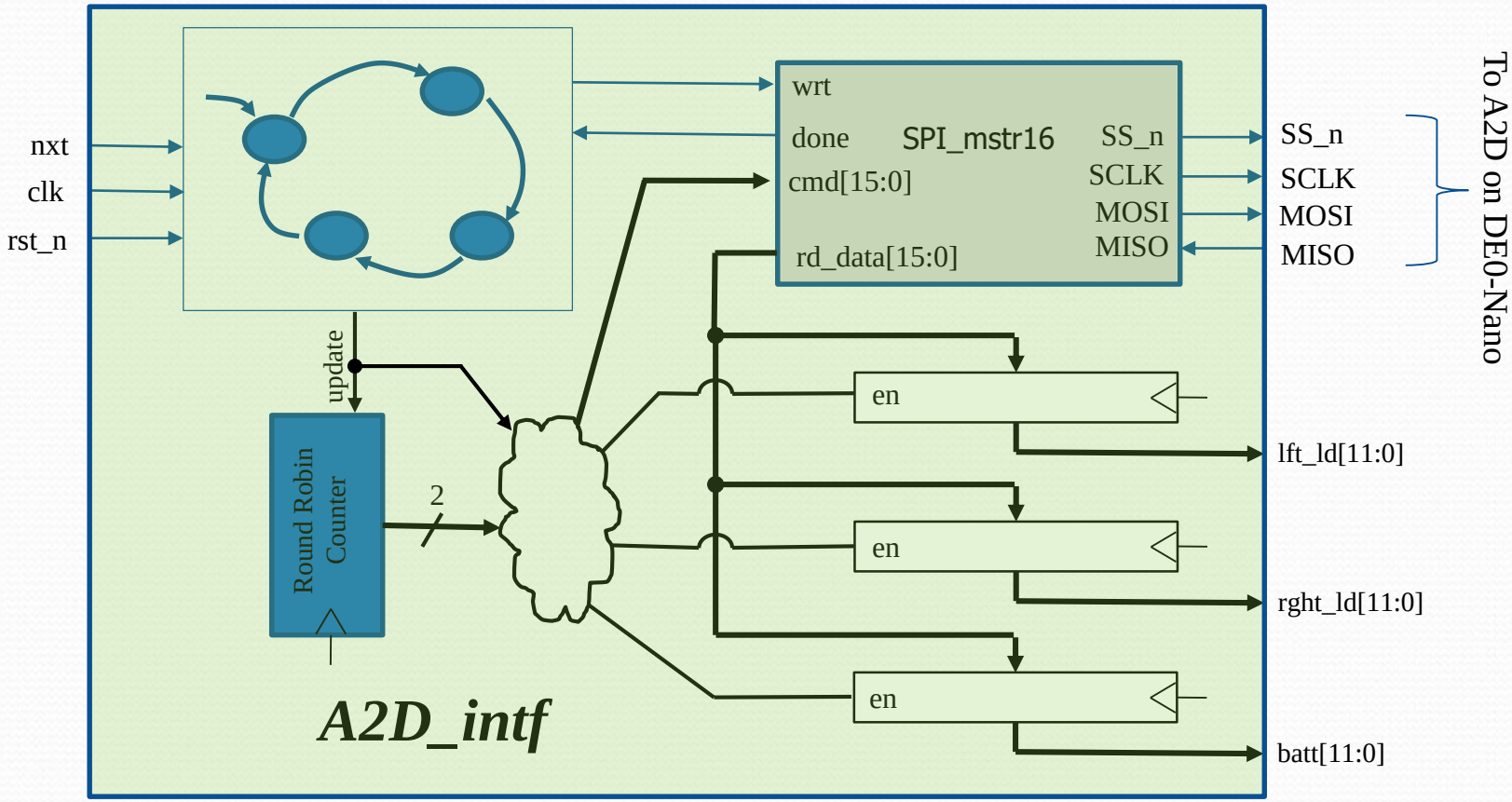
Exercise 17: A2D Intf Design and Test Bench (HW5 Prob3)

In HW4 you produced a SPI master (SPI_mstr16.sv). We are now going to use that block to make a block that does round robin conversions on channels 0,4,5.

You will be producing a module called **A2D_Intf.sv** with the following interface:

Signal:	Dir:	Description:
clk,rst_n	in	50MHz clock and active low asynch reset
nxt	in	Initiates A2D conversion on next measurand
lft_ld[11:0]	out	Result of last conversion on channel 0 (left load cell)
rght_ld[11:0]	out	Result of last conversion on channel 4 (right load cell)
batt[11:0]	out	Result of last conversion on channel 5 (battery voltage)
SPI Interface	Out/ in	SS_n, SCLK, MOSI, MISO of a SPI interface. Comes from copy of SPI_mstr16 embedded into this unit.

Exercise 17: A2D Intf Design and Test Bench



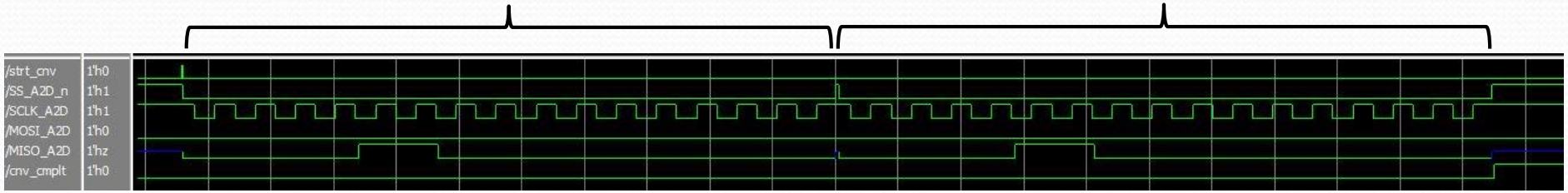
The SM will sit idle till it is told to perform a conversion (**nxt** asserted). Thn it will kick off two SPI transactions via **SPI_mstr16**. The first SPI transaction determines what channel to convert, and the second SPI transaction reads the result for that channel. The round robin counter is then incremented, and on the nxt request it will convert the next channel in the sequence.

You also need 4 holding registers to hold the respective results. The round robin counter determines where to store the results as well.

Exercise 17: A2D Intf Design and Test Bench

First 16-bit SPI transaction specifies The channel to perform conversion on. Data returned on MISO is junk.

Second 16-bit SPI transaction the data sent over MOSI does not really matter, just reading result over MISO.



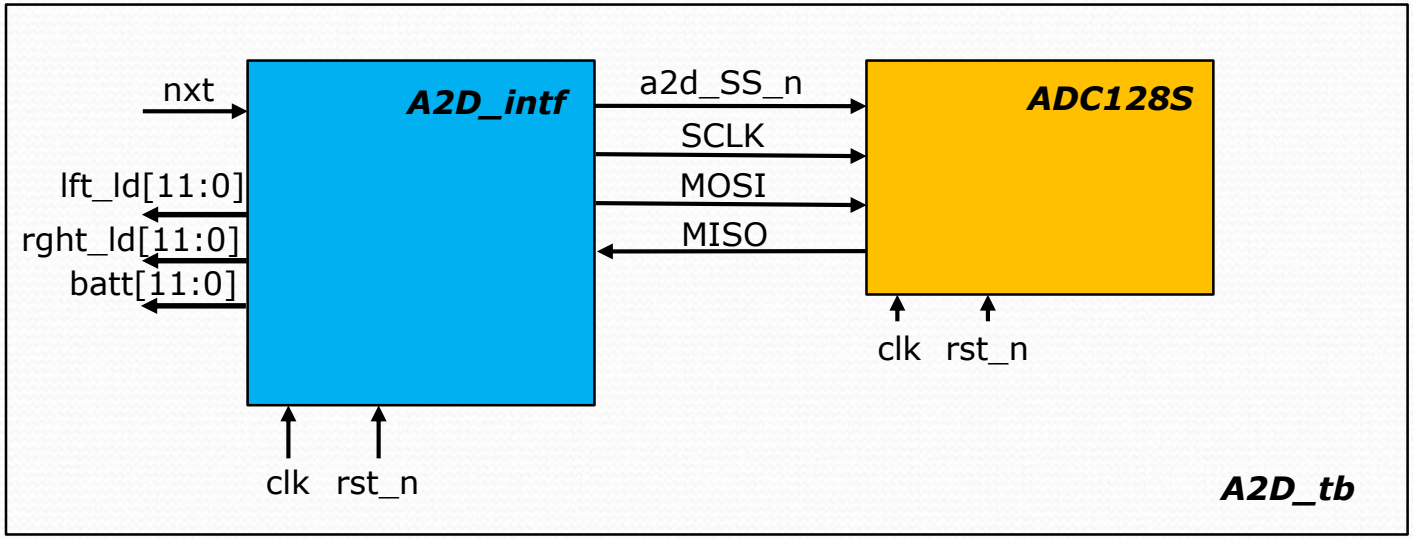
Our use of the A2D converter will involve two 16-bit SPI transactions nearly back to back (separated by 1 system clock cycle).

The first transaction here is sending a 0x0000 to the A2D over MISO. The command to request a conversion is {2'b00,channel[2:0],11'h000}. The upper 2-bits are always zero, the next 3-bits specify 1:8 A2D channels to convert, and the lower 11-bits of the command are zero. Therefore, the 0x0000 in this example represents a request for channel 0 conversion (channel 0 is the left load cell in our application).

For the next 16-bit transaction the data sent over MOSI to the A2D does not matter that much. We are really just trying to get the data back from the A2D over the MISO line. The data we get back in this example is 16'h0C00. Of course since it is a 12-bit converter only the lower 12-bits (12'hC00) is valid.

Exercise 17: A2D Intf Design and Test Bench

A model of the A2D converter is provided on the course website (**ADC128S.sv**). Download this and make a test bench that incorporates your A2D_intf and ADC128S.



NOTE: ADC182S will return 0xC00 for the first reading, and then 0x010 less for every subsequent reading. i.e. 0xBF0, 0xBE0, … if the channel was 0. It will return 0xC01 and then 0xBF1, … if the channel read was 1. If first channel read was 0 and 2nd channel read was 2 answers would be: 0xC00, 0xBF2

Exercise 17: A2D Intf Design and Test Bench

There is no dropbox for this exercise. In Exercise 18 you will demonstrate your A2D_intf on the DE0-Nano.