Bus Interface Design Problem (35 points total)

For this problem, you need to complete the bus interface to the dipswitches, Bar Leds and 7-Segment Leds shown in the High Level schematic. The problem is similar to that of lab 3, except we will use a low power FPGA family that performs somewhat slower then the FPGA used in the lab. Be sure to read the timing specifications carefully.

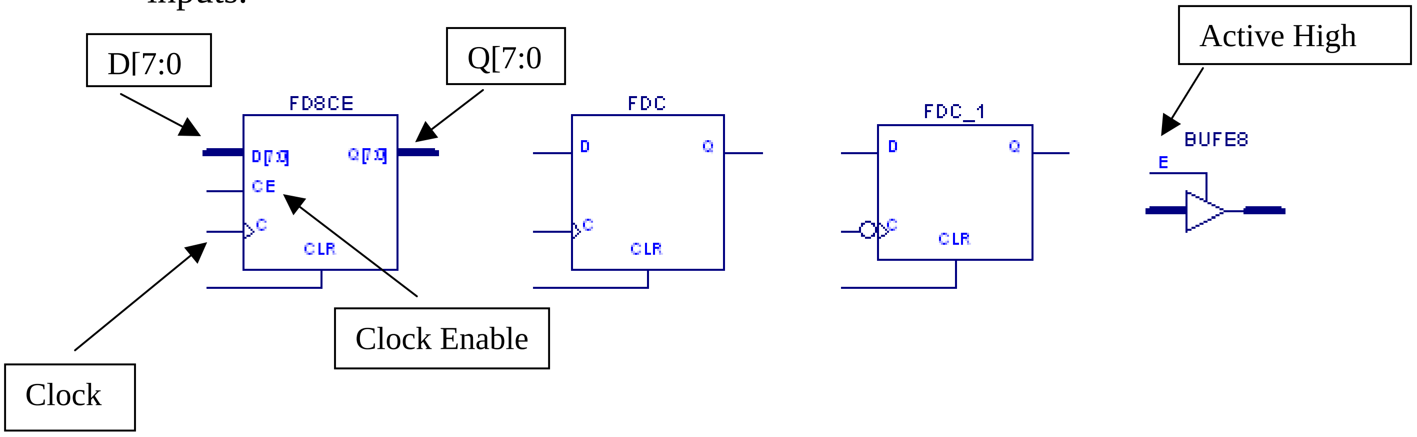
Power PC Timing Specifications
TA* and Data Setup: ¼ clock cycle
TA* and Data Hold: ¼ clock cycle

FPGA Timing Specifications
TRI-State Buffer Propagation Delay: negligible
LED Register Data Setup: 2 clock cycles
LED Register Data Hold: ¼ clock cycle

Memory Map
Switch Location (Word Addressable): 0x3100000
Bar Led Location (Word Addressable): 0x3200000
7-Segment Display (Word Addressable): 0x3200000

Devices Allowed in the Design
You will be asked to complete various parts of the bus interface design. The following components are allowable.

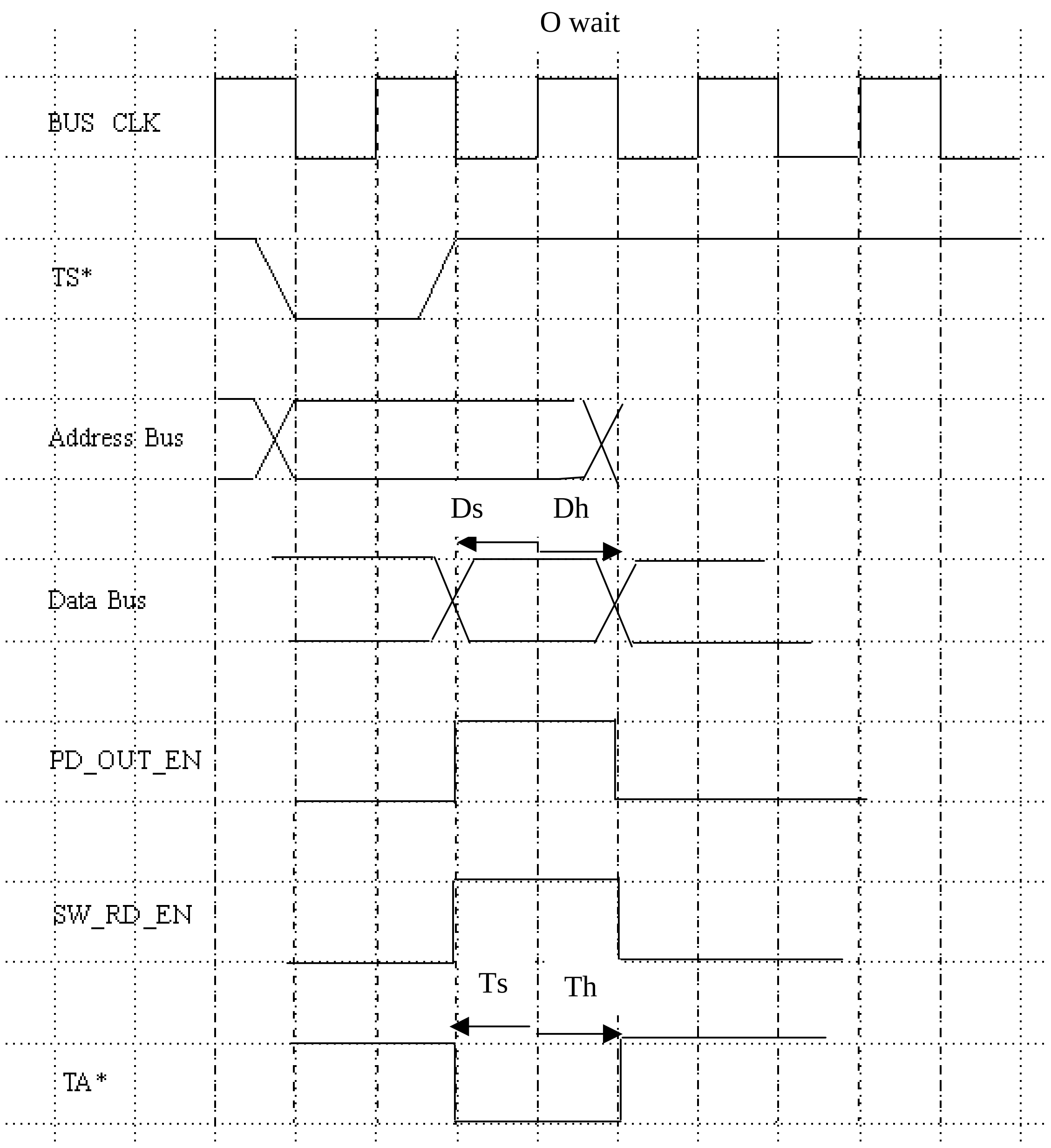
Inverters
AND, NAND, OR, NOR of any size. You may use inverting bubbles on inputs.



Timing Study
In lab, we discussed that the design process should start with a timing study. We started by addressing the minimum requirements. We then considered a synchronous solution that would meet the minimums with a safety margin.

Assignment 1 (10 points)
Complete the following read cycle timing diagram for a synchronous solution. You should use **no** more WAIT states then necessary.

Show the TA*, and processor data setup and hold times **your design will provide**. Illustrate the times as intervals on the timing diagrams.

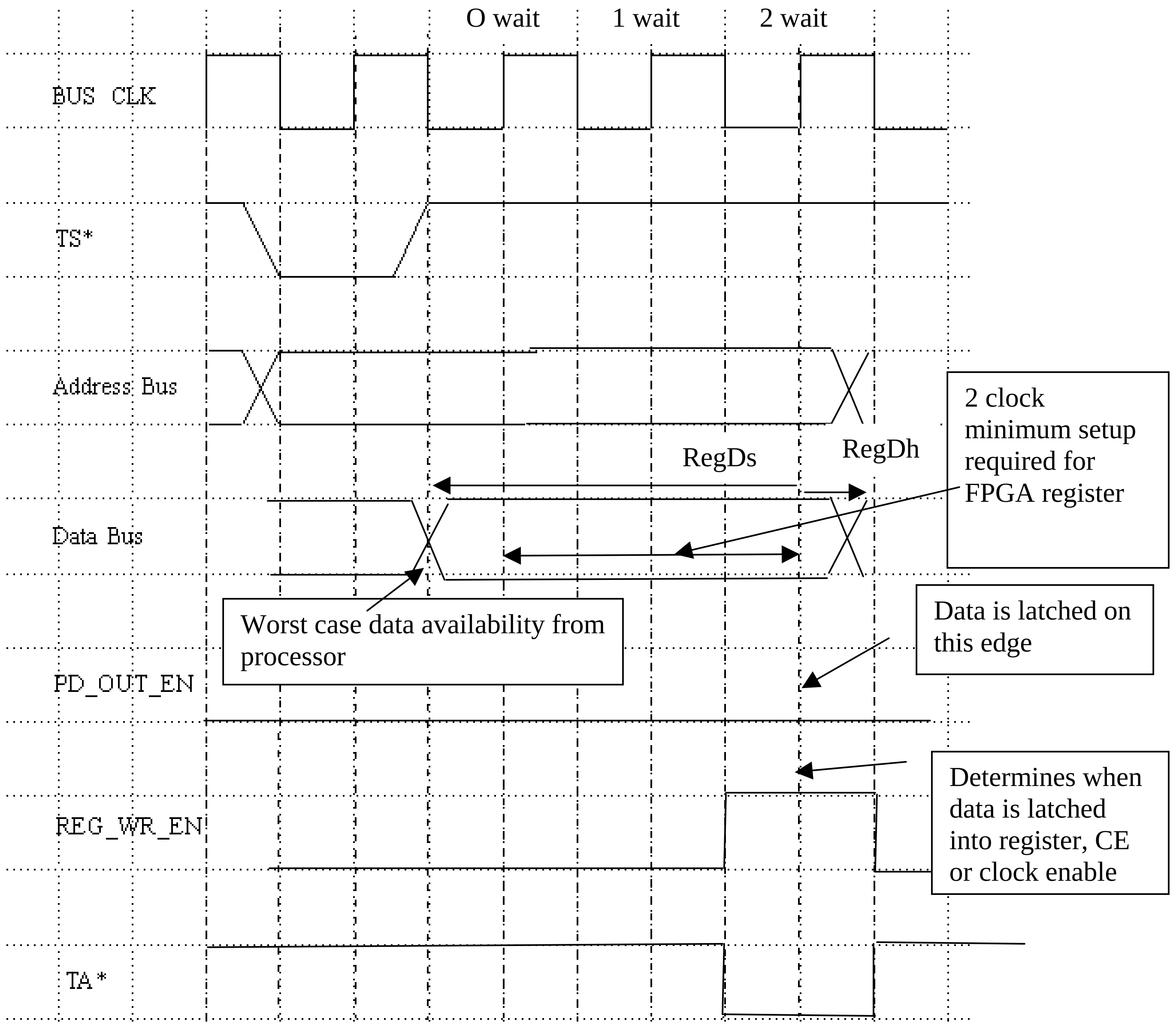


Assignment 2 (10 points)

Complete the following write cycle timing diagram for a synchronous solution. You should use **no** more WAIT states then necessary.

Show the LED registers data setup and hold times **your design will provide**. Illustrate the times as intervals on the timing diagrams.

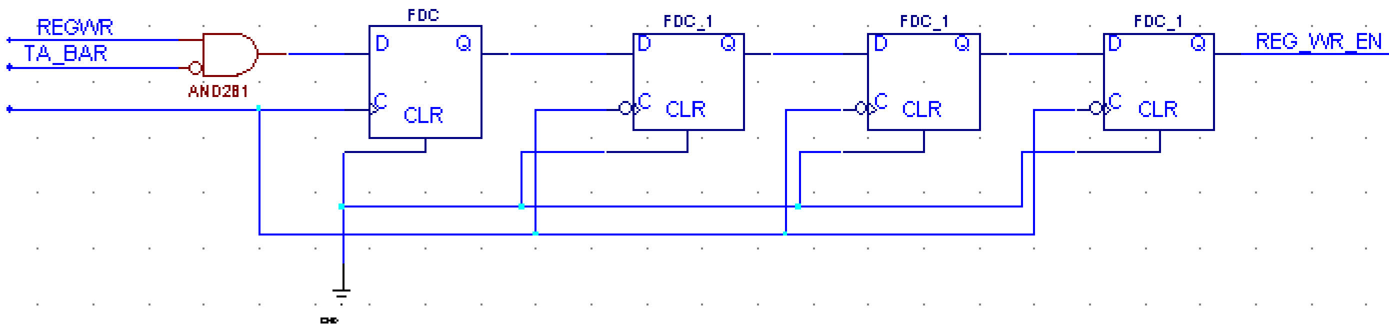
Complete LED registers connections on the High Level schematic. Be sure to show all the connections clearly.



O wait
1 wait
2 wait
BUS CLK
TS*
Address Bus
Data Bus
PD_OUT_EN
REG_WR_EN
TA*
RegDs
RegDh
2 clock minimum setup required for FPGA register
Worst case data availability from processor
Data is latched on this edge
Determines when data is latched into register, CE or clock enable

Assignment 3 (5 points)

Provide the logic in schematic form in the space below for the **REG_WR_EN** signal in the IO control macro. **You DO NOT have to provide logic for the other control signals**. You **DO NOT** have to provide the address decode. Notice it is provided to the macro. Clearly label all signals.



There are a few other solutions that will work. You received full credit if you provided the correct shift and conditioning.

Assignment 4 (**5 points**)
In this design, we want to provide a means to read LED registers as well as writing them.

Modify the High Level schematic for the paths and hardware required to implement reading the LED registers. Label all connections clearly. Assume that the BAR led is read on bits 24-31 and the 7-Segment is read on bits 17-23.

Assignment 5(5 points)
If the Led registers were upgraded to be byte write addressable:

What input and output control signals do you have to add to the address decoder?

Input: TSIZ0, TSIZ1   Output: REG_WR_byte2, REG_WR_byte3

This should really be 0x320000x. Where x = 0b00xx and x is don't care. A31 and A30 need to be decoded for byte addressing. I omitted this in the grading.

Assuming the address 0x3200000 is decoded as AD32 (active high), provide the **Boolean** expressions for the LED registers byte write control signals. Use ***** for AND, | for OR and **~** for inversion.

You **do not** have to provide any modifications to the IO_Control Macro.

REG_WR_byte2 =

Provides byte addressing

(A30 * ~A31 * TSIZ1 * ~TSIZ0 * AD32 * ~RDWR) |
(~A30 * ~A31 * ~TSIZ1 * ~TSIZ0 * AD32 * ~RDWR)

Maintains Word Addressing

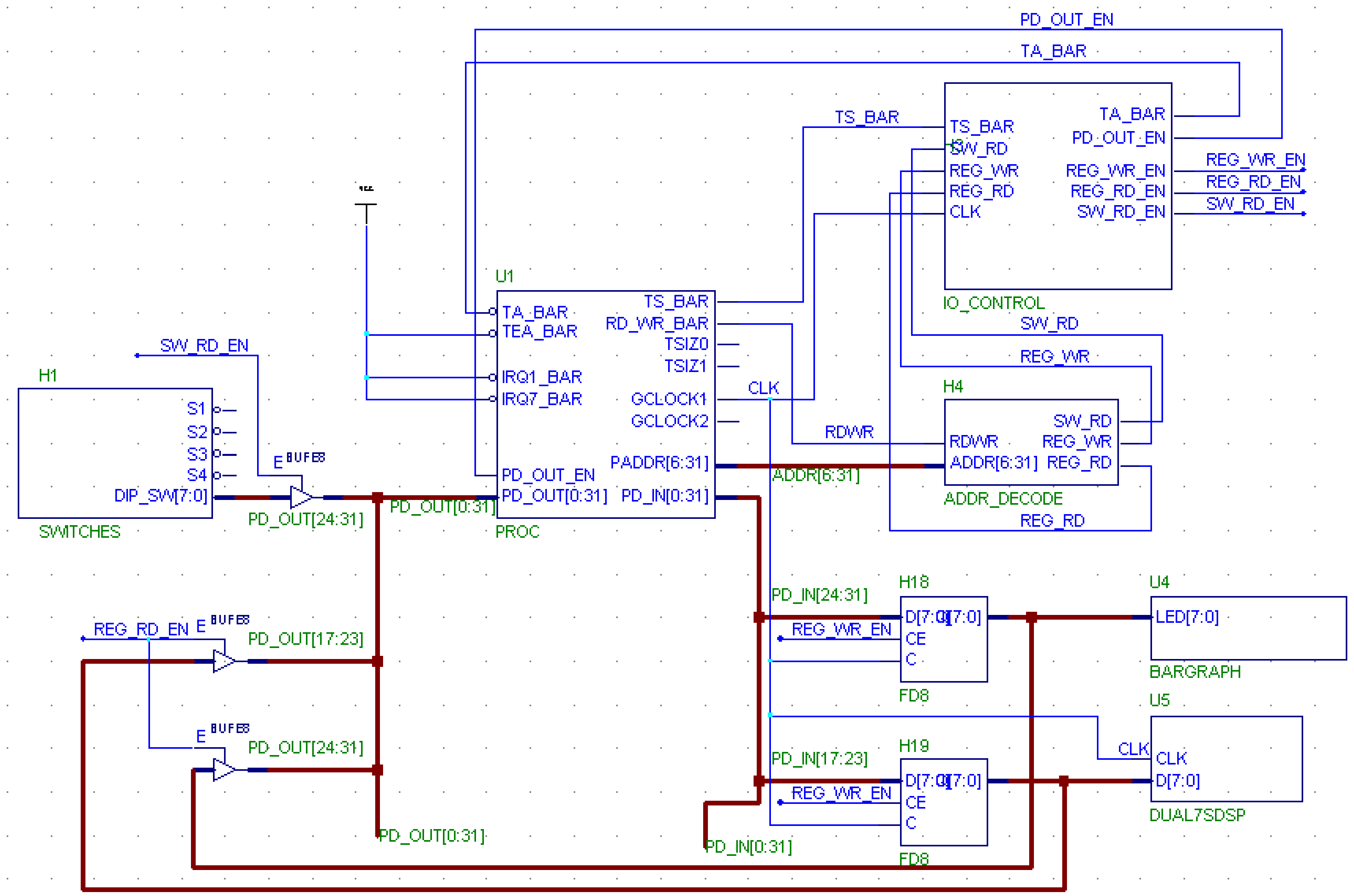
REG_WR_byte3 =

(A30 * A31 * TSIZ1 * ~TSIZ0 * AD32 * ~RDWR) |
(~A30 * ~A31 * ~TSIZ1 * ~TSIZ0 * AD32 * ~RDWR)

There are no registers for bytes 0 and 1 so they are not decoded. Half word addressing of the byte 2 and byte 3 register was not asked for, so it is not provided.

Assignment 5 (additional space)

HIGH LEVEL SCHEMATIC



PD_OUT_EN
TA_BAR
TS_BAR
TS_BAR
SW_RD
REG_WR
REG_RD
CLK
TA_BAR
PD_OUT_EN
REG_WR_EN
REG_RD_EN
SW_RD_EN
REG_WR_EN
REG_RD_EN
SW_RD_EN
IO_CONTROL
U1
TA_BAR
TEA_BAR
IRQ1_BAR
IRQ7_BAR
PD_OUT_EN
PD_OUT[0:31]
TS_BAR
RD_WR_BAR
TSIZ0
TSIZ1
GCLOCK1
GCLOCK2
PADDR[6:31]
PD_IN[0:31]
PROC
SW_RD_EN
H1
S1
S2
S3
S4
DIP_SW[7:0]
SWITCHES
E
BUFE8
PD_OUT[24:31]
PD_OUT[0:31]
SW_RD
REG_WR
CLK
H4
RDWR
RDWR
ADDR[6:31]
SW_RD
REG_WR
REG_RD
ADDR[6:31]
ADDR_DECODE
REG_RD
PD_IN[24:31]
H18
D[7:0]
Q[7:0]
CE
C
REG_WR_EN
FD8
U4
LED[7:0]
BARGRAPH
REG_RD_EN
E
BUFE8
PD_OUT[17:23]
E
BUFE8
PD_OUT[24:31]
U5
CLK
CLK
D[7:0]
DUAL7SDSP
PD_IN[17:23]
H19
D[7:0]
Q[7:0]
REG_WR_EN
CE
C
FD8
PD_OUT[0:31]
PD_IN[0:31]