Questions from page 7

Please answer the following in a new word document.

a. What is banking? How many banks are on this FPGA? Why is having different banks useful?
   Banking is used to create buffers between the core voltage and outside elements operating at differing voltages. They are used to provide logic level conversion between separate devices running at varying voltages, allowing for communication between them. There are 8 banks on the Altera Max 10.

b. There is an I/O standard setting that you can change. What's the difference between 3.3V TTL and 3.3 V LVCMOS?
   Both are transistor technologies for driving voltages, generally operating within the same ranges. The TTL type can function with most other transistor technologies, except for the original non-LV CMOS type. Original CMOS and its LV varieties can communicate with each other more easily.

c. What does changing the current strength do? Why would you want to increase or decrease the current strength?
   Changing the current strength would affect the voltage being supplied to a device. It might be done to bring the sampled voltage higher or lower into a desired range for a high or low logic level.

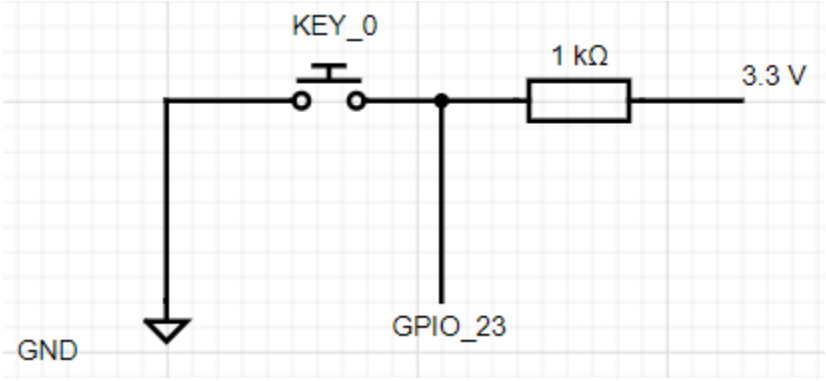
d. What does changing the slew rate do? Why would you want to change it?
   Changes the speed at which the logic level transitions. You might change increase it to meet setup/hold times; you might decrease it to reduce the noise of the signal.

e. Do you always want the most current and highest slew rate? Why or why not?
   Neither necessarily; higher slew rates often result in signal spikes on logic level transitions followed by oscillations, making for inaccurate voltage sampling.

   Higher current may result in voltages beyond the acceptable high voltage range of a device or part, leading to inaccurate sampling as well as potentially dissipating too much power and damaging something.

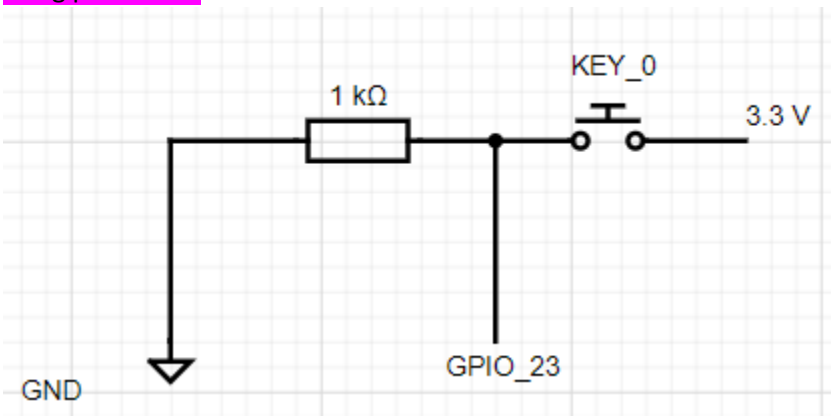
Questions from page 9 and 10

1. Questions: Please draw how a single button is connected to the FPGA using a pull-up resistor.



A pullup resistor is used to achieve active LOW behavior: when KEY_0 is pushed, the signal sent to GPIO_23 is LOW.

2. Question: What is the difference between pullup and pulldown? Please draw a single button using pull-down.



A pulldown resistor achieves the opposite: when KEY_0 is pushed, the signal sent to GPIO_23 is HIGH. Both kinds resistors prevent the input voltage from shorting to ground.

3. Question: Does the MAX10 FPGA have both internal pullups and pulldowns available? Where in the datasheet does it describe this functionality? Link or snipping is fine.

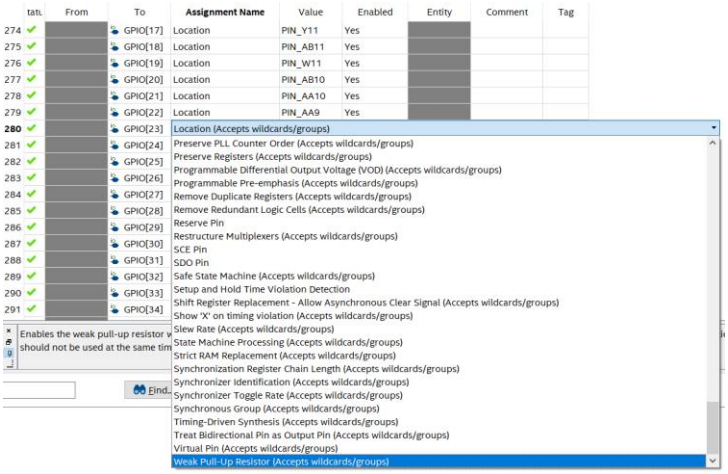
From the data sheet:

Internal Weak Pull-Up Resistor

All I/O pins, except configuration, test, and JTAG pins, have an option to enable weak pull-up.

There is no mention of pulldown resistors in the datasheet.

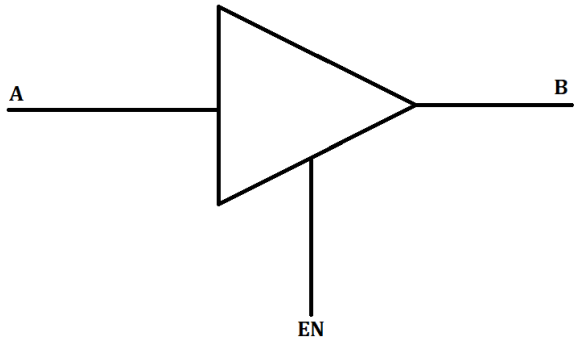
4. How would you enable an internal pull-up or pull-down?



Questions from page 11

1. Please draw one tristate buffer below and explain in your own words what tristate buffer is, and how tri-state works. This should have been covered in CST 133 and CST 162, but you can use the internet if you wish. **Just make sure that your answer is in your own words or drawing.**

    1. Draw the tristate buffer



    2. Explain what is tristate?

    Tristate is the third possible state in a binary system: 0 (low), 1 (high), and Z meaning the signal is disconnected or otherwise at high impedance.

    3. How does tristate work?

    In the diagram above, the EN signal enables the A signal to output to B. If the EN is set low (or high if active LO) then B is effectively disconnected from the greater circuit containing A and EN.

    4. What could it be used for?

    It could be used to exchange communications on a shared line, with each party having their own tristate buffer as well as controlling the enable signal of the opposite party's buffer. Party A effectively mutes party B and vice-versa when they begin to talk.

2. So, how does it work in Verilog? Copy and paste the code below and compile.

```
65  //assign LEDR[6:0] = ARDUINO_IO[8:2];
66
67   assign LEDR[6:0] = SW[9] ? 7'bzzzzzzz : ARDUINO_IO[8:2];
68
```

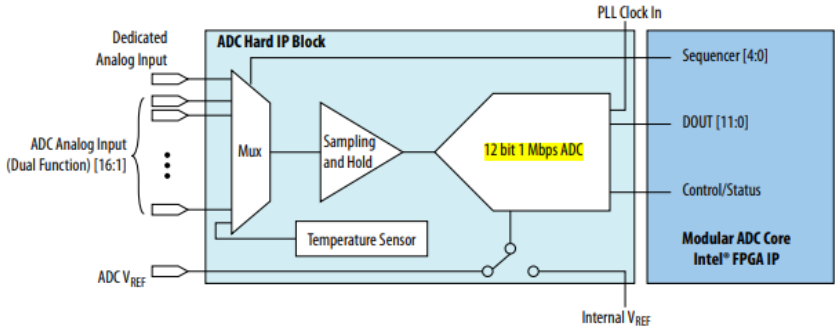
3. Please explain what line 67 is doing.

Line 67 is a tristate buffer which uses SW_9 as an enable signal. When SW_9 is high, the LEDs are effectively disconnected from the device. When SW_9 is low, the signals from the seven ARDUINO_IOs are passed along to the LEDs.

4. Ok, now demonstrate to your instructor tristate functionality.

Questions from page 12

1. Answer the following
   a. How many bits ADC is the MAX10 built in ADC, and what is the sampling rate for this ADC? Give me a link to where you found this.



pg 12 of the ADC User Guide

   b. How many channels does the MAX10 device on the DE10-Lite board support? Give me a link to where you found this.

Two dedicated channels and 16 dual function channels.

Package	Pin Type	ADC Channel Counts Per Device					
		10M04	10M08	10M16	10M25	10M40	10M50
M153	Dedicated	1	1	—	—	—	—
	Dual function	8	8	—	—	—	—
U169	Dedicated	1	1	1	—	—	—
	Dual function	8	8	8	—	—	—
U324	Dedicated	1	1	1	—	—	—
	Dual function	16	16	16	—	—	—
F256	Dedicated	1	1	1	2	2	2
	Dual function	16	16	16	16	16	16
E144	Dedicated	1	1	1	1	1	1
	Dual function	8	8	8	8	8	8
F484	Dedicated	—	1	1	2	2	2
	Dual function	—	16	16	16	16	16
F672	Dedicated	—	—	—	—	2	2
	Dual function	—	—	—	—	16	16

pg 6 of the ADC User Guide

   c. If an ADC is 10-bit resolution and the voltage range is 10 V, what is the step size of each value?

11_1111_1111 = 1023 => 10 V / 1023 = 0.009775 ~ 9.8 mV step size

Questions from page 18

1. Record the values (Remembering that you can't see the two LSB bits- just set them to zeros.)
    a. Right movement of joystick maximum value: [issues with board]
    b. Centre (idle) joystick value for X axis: 01_0110_1111
    c. Left movement of joystick maximum value [issues with board]
    d. Centre (idle) joystick value for X axis: 01_0110_1111
    e. Up movement of joystick maximum value 10_1011_1011
    f. Down movement of joystick maximum value 00_0000_0001
2. Demo to instructor the joystick functionality (indicated using LED and switch 6)

3. Question: What is the step size of each increment assuming 12 bit resolution and 5 V range?
    1111_1111_1111 = 4095 => 5V / 4095 = 0.001221 ~ 1.22 mV

4. Question: Jostling the board slightly will change the values slightly on the LSB side. Pressing the buttons will also cause the value seen on the LEDs to change. Why is this the case? Think in terms of step size and noise.
    The joystick operates based on motion, so it would be temperamental to movement acting on the potentiometers, including vibrations from a button being pressed. If the motion causes the joystick's potentiometer to effect a change of a little over a thousandth of a volt, it will register as a change in the voltage, causing the LED values to recalculate.

5. Question: It seems that the ADC can be somewhat noisy. If we were feeding the joystick input into another system (such as controlling a cursor or something), this could prove to be noisy. What ways can you think of to eliminate or filter noise?
    Changing the step size to be larger would make it so that a greater amount of change needs to be registered by the system before recalculating the incoming voltage. The step size could be increase with a higher incoming voltage or a smaller resolution (10 bits instead of 12).

6. Question: IP cores are pre-fabricated modules which allow us to access functionality of the board. Convenient. Any disadvantages to using IP cores in our design? Please highlight two disadvantages.
    1. Can't change the bit resolution of the ADC, it is only ever 12 bits.
    2. From pg 8 of the ADC user guide:

    > The ADC in dual supply Intel MAX 10 devices can measure from 0 V to 2.5 V. In single supply Intel MAX 10 devices, it can measure up to 3.0 V or 3.3 V, depending on your power supply voltage.

    The ADc can't measure a voltage beyond 3.3 V.

Note from page 19

1. Go to Project > Clean Project.
2. Zip up your project and submit to Canvas.