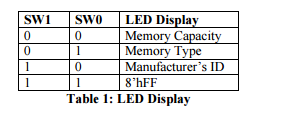
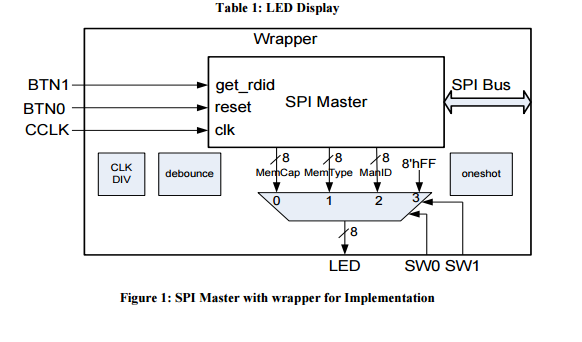
**Introduction**

In this Homework assignment, we will be taking our code from Homework 4 and put it on the FPGA.

Key Concepts:

* Debounce
  + Debouncing input get\_Rdid
* One-shot
  + One-Shotting the input get\_rdid
* Working with the LED Screen



* Implementing a wrapper class
* 

**DO NOT**

* Free-run the spi clock
* Continuously assert the chip select of the memory

**Hints/Recmmendations**

* Instead of heavily modifyin HW4 you create a wrapper that instantiates the SPI master and any clock dividers debouncers or one-shots.
  + This could mean I create a separate module/file for Debounce, CLK DIV, and one-shot
* FPGA Config
  + Use button 0(The north button) for the reset input
  + Use button 1(the east button) for the get\_rdid input.

Deliverables:

Report (50pts)

1. Lab report according to the format – 15 pts

2. Code for your SPI Master, Wrapper, and other modules incorporated – 25 pts

3. Print-screen of– 10 pts

a. The Chipscope waveform with the signals in 5.a; 5.b; 5.c;

b. The synthesis, implement, generate-program steps completed (on left side), and design summary which includes, number of slices, warnings for each step.

Demo (50 pts)

4. A Working demonstration

5. Chipscope plot of:

a. The debounced and possibly one-shot’d get\_rdid signal

b. The SPI bus (4 signals) – spiclk, spimiso, spimosi, prom\_cs\_n

c. The lower 3 bits of Memory capacity

Notes:

1. Full credit for the demonstration will be given if you can show:

a. When BTN1 (get\_rdid) is pressed the SPI bus shows a correct sequence.

b. The LED’s clear when BTN0 (reset is pressed).

c. The correct data appears periodically on the LED’s

2. Demonstrations are due by the end of class on the due date.

3. Late demonstrations will be accepted 1 week later only with a penalty of 50%

4. Lab report is due one week after the demo.