**Design Name :** A Key Sequence Reader Example

**Objective :**

Learn how to store inputs in an array/vector.

**Assignment 1:**

We would like to design a key-sequence tracker, but we have only two keys. That is, users will enter a binary sequence using these keys which represent binary 0 and 1.

1. Use two on-board buttons (East=0, South=1).
2. When the user presses East button, the value 0 is shifted into the LSB bit of an array of length 8-bits, shifting previous values to the left (towards MSB). MSB will be discarded.
3. Same actions will be taken when the user presses South button with inserted bit value of 1.
4. Last 8 bits will be displayed on LEDs.
   Example : User presses S,E,S,S,E,E,S,E, LEDs will show 10110010. Then if user presses E, LEDs will show 01100100.

Note: You need to use key-debouncers. Array can be either a `std_logic_vector(7 downto 0)` or an `array(7 downto 0) of std_logic`.

**Assignment 2:**

Enhance your design so that, when Center button is pressed your code checks if the last 8 bits entered match the preset constant pattern. In case of match all LEDs light up, otherwise all LEDs are turned off.

**Assignment 3:**

Enhance your design so that, when match occurs, all LEDs blink at 2 Hz.

**Assignment 4:**

Enhance your design so that all four buttons are used to enter the key sequence. Now a single bit array (or vector of bits) is not enough to store key value. Assume that E=00, S=01, E=10 and N=11. Think of a way to display a 16 bit value on 8 LEDs (use a switch for example).

**Homework :**

Describe the changes required in the design for a 10-key keyboard and longer key-sequences.