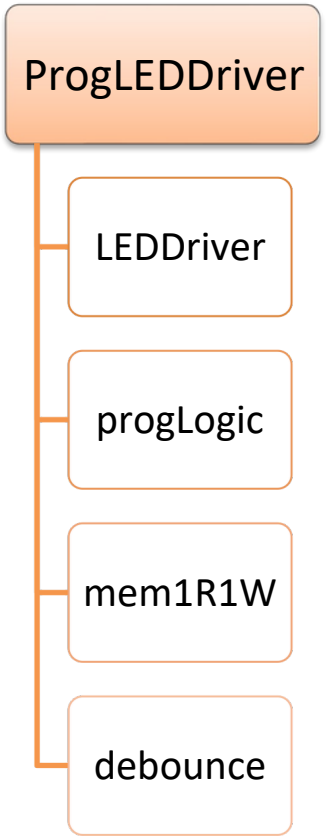
Programmable LED Driver

Hierarchyof ProgLEDDriver



Top module of the design and it only contains instantiation of the other modules. You will use existing code for this module.

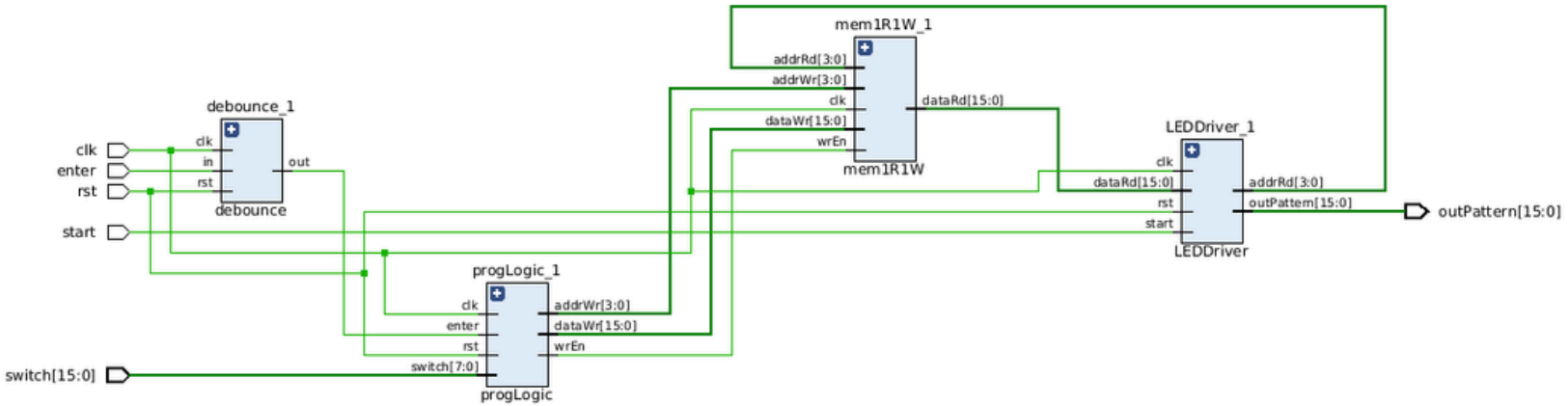
This module reads values from RAM and displays them on LEDs. It should start displaying after the start signal. You will implement this module.

This module is for filling RAM using switches and a push button. You will implement this module.

This module is RAM module.
You will use existing code for this module.

This module is to eliminate bounce problem.
You will use existing code for this module.

Block Diagram of progLEDDriver



Design I/O:

clk : 1 bit input for clock

rst : 1 bit input for reset

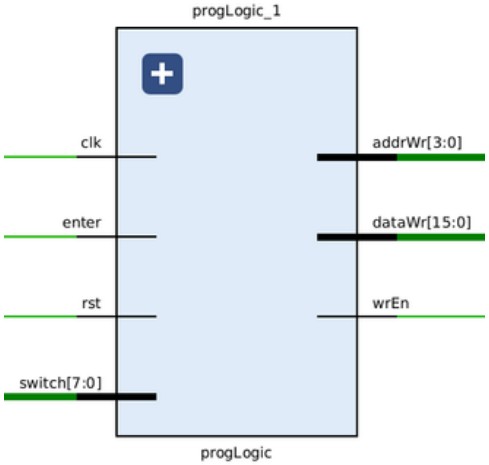
switch : 16 bits input comes from switches as data in

enter : 1 bit input comes from a push button when data is ready on switches

dataWr : 16 bits data output to RAM

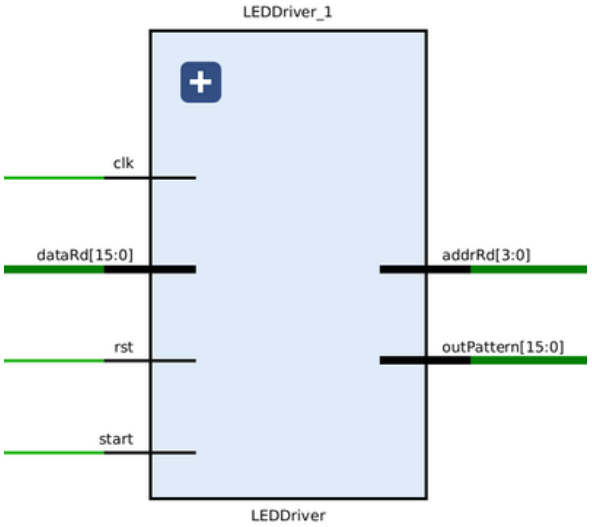
addrWr : 3 bits address output to RAM

wrEn : 1 bits output as write enable to RAM



Design I/O:

clk : 1 bit input for clock
rst : 1 bit input for reset
start : 1 bit input comes from a push button when data is ready to display
dataRd : 16 bits data input from RAM
addrRd : 3 bits address output to RAM
outPattern: 8 bits of output to LEDs



Design Behavior:

- The aim of this module is to fill RAM and read it. RAM entries have 16 bits of length, and we will use 16 bit switches to fill it.
- We will use an enter signal, which comes from a push button, as data ready signal.
- Meanwhile, the wrEn signal should be 1 when we want to write data to RAM. Otherwise, it should be 0.
- When you press the start button (another push button), the design should start displaying the data stored in the memory.