The purpose of this first homework assignment is to ensure an understanding for Register Transfer Level (RTL) Design through the example of constructing a Parallel to Serial Circuit (P2SC) for an 8-bit input data stream.

**Datapath**

The following diagram is a visual representation of the Datapath of the 8-bit P2SC:



The only required hardware components are an 8-bit data register, a tri-state gate (which can be modeled by a MUX) for output enabling, and a 3-bit up counter to address the data register. The interoperability of these components will be explained in the Controller description of this logic circuit. The following is a key to distinguish all colors used in the model:

**Red** text - Components (actual hardware) of the Datapath  
**Blue** text - Signals issued by the Controller to control the Datapath

**Green** text - Signals that are received from or sent to an external source**Controller**

The following diagram is a visual representation of the Controller of the 8-bit P2SC:



The following Controller layout simply defines all necessary logic needed to make the 8-bit P2SC operate correctly. The following section describes all different states located within the controller, along with all signals that are being set. Note that all Controller operations are ***clk*** dependant:

@ **‘Idle** state:

> Controller is waiting for ***dataready* = 1** for one ***clk*** pulse

> Next state is **‘Init**

***clr*** = 1, ***count\_en*** = 0, ***ld\_en*** = 0, ***out\_en*** = 0

***ready*** = 1

@ **‘Init** state:

> Controller enables data to be read into **8-bit register** in the **DataPath**

> Next 8-states are for serial data output

***clr*** = 1, ***count\_en*** = 0, ***ld\_en*** = 1, ***out\_en*** = 0

***ready*** = 1

@ **‘d1, ‘d2, …, ‘d8** states:

> Controller enables counter & output enable to access data in the register > Increment the counter and the ***addr*** reference for 8-clock cycles

> Loop back to **`Idle** state when done, and wait for ***dataready = 1*** again

***clr*** = 0, ***count\_en*** = 1, ***ld\_en*** = 0, ***out\_en*** = 1

***ready*** = 0

With the proper integration of the **Datapath** and **Controller** implementations shown above, the Parallel to Serial Circuit (P2SC) is realized, and can easily be modeled in a Hardware Description Language (HDL) like Verilog or VHDL.

This concludes the analysis for Homework 01.