Lab session #05

October 24, 2024

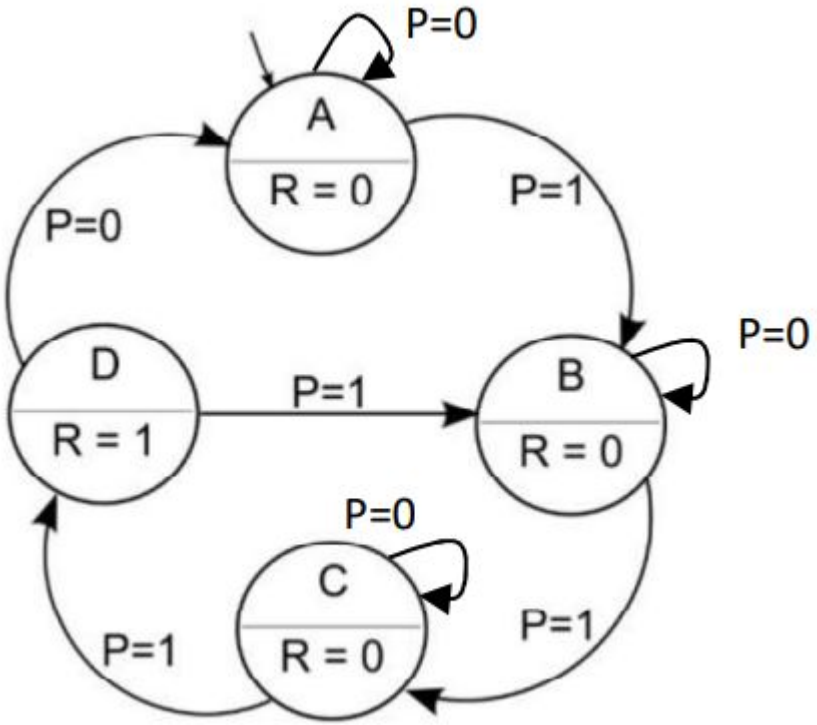
Michelangelo Barocci, michelangelo.barocci@polito.it

Politecnico di Torino,

PhD Student @ EDA Group - DAUIN

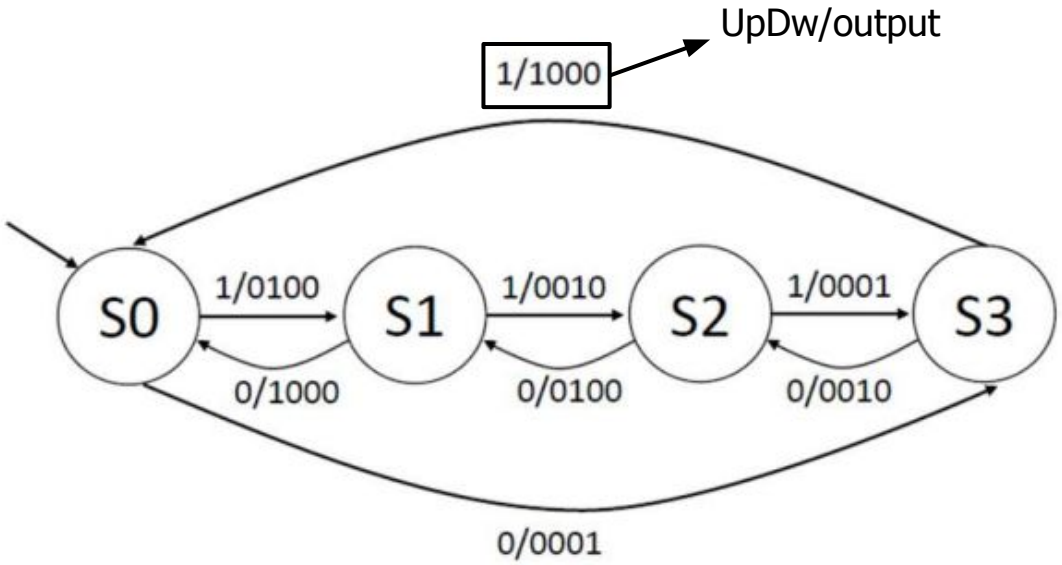
Dpt. of Computer and Control engineering (DAUIN)

FSM #1



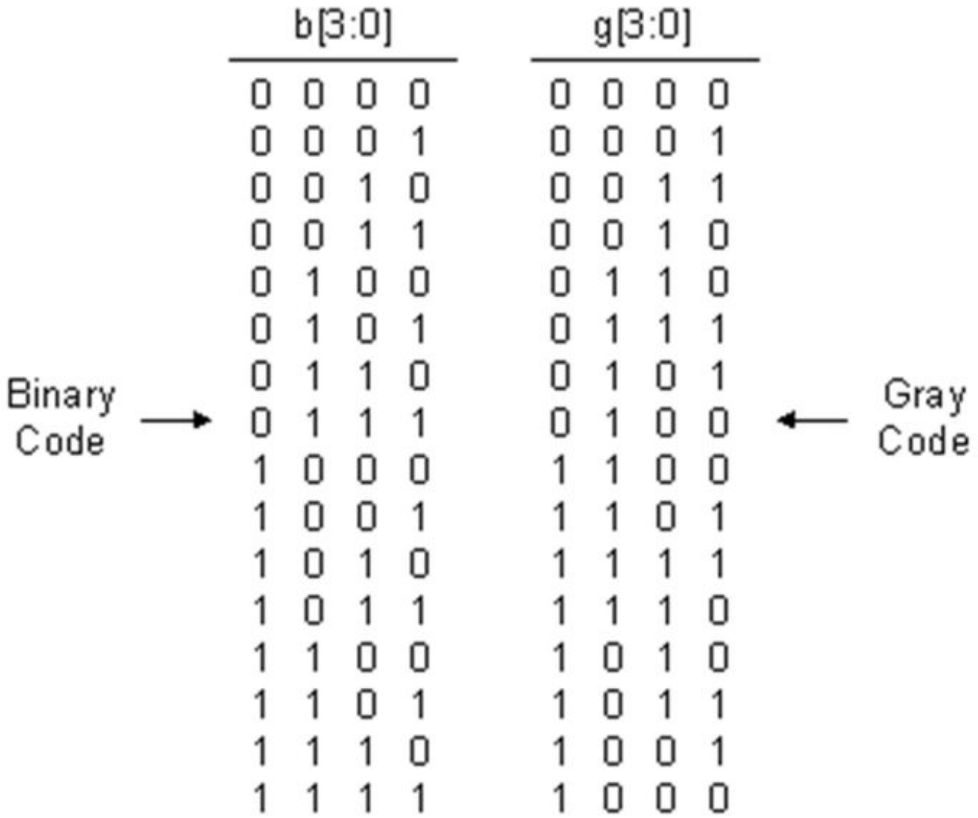
- Design a Finite State Machine with these characteristics:
  - one input signal P to control the next state's decision
  - one output signal R (Moore output)
  - asynchronous reset
- Vary the design by:
  - Changing the type of state encoding (minimum length and one hot)
  - instantiating the FSM with 2 PROCESS and 1 PROCESS
- What changes?
- Write a testbench to test your FSM design on relevant inputs. Use a file to import input vectors and read expected output values.

FSM #2



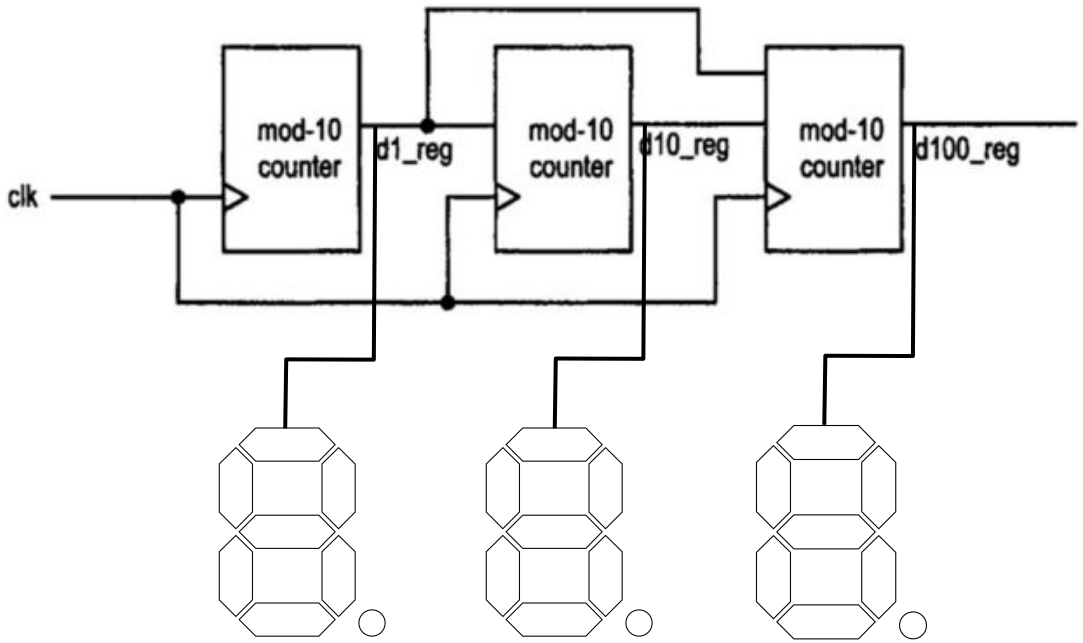
- Design a Finite State Machine with these characteristics:
  - one input signal UpDw
  - one output signal (Mealy output)
  - asynchronous reset
- The state decision is made by reading the signal UpDw, that determines also the current state output. Follow the state diagram in the picture.
- To test the design use an external file to feed the input vectors to the testbench

Gray Counter



- Design an FSM-based N-bit Gray counter with asynchronous reset
- Remember how we implemented a Gray counter in Lab#03: now find a smart way to store the Current and Next state signals
- test your design with a testbench properly written for N=4

Decimal counter



- Design a BCD counter 0-999 by cascading three mod-10 (4-bit) counters: each counter represents a decade (100, 10 and 1) and has its own increment rules: a digit increments depending on the previous digit's current value
- Use a FSM-like approach: define the current and next digit of each counter as a finite state
- Test your design with a testbench and verify that all transitions are enabled correctly.