Problem 1. Design a synchronous 5-bit one-hot code rotator with CE and reversibility (UD_L) using a FSM strategy. The output codes generated must be: 00001, 00010, 00100, 01000, 10000, 00001 ...

a. Draw the symbol and name its inputs and outputs, Draw its function table. Sketch the state diagram and plot an example of a timing diagram.
b. Draw the FSM structure consisting of of CC1, CC2 and state register, indicating the name of the inputs and outputs. How many D-FF are necessary in this application if coding internal states in binary? How many D-FF are necessary if coding internal states in one-hot?
c. Draw the internal structure of the state register if the system is coded in one-hot
d. Write the truth tables for CC1 and CC2
e. Draw the flow chart for CC1 and its translation to VHDL code.
f. If the $t_{CO}$ (CLK to output propagation time) of a D-FF is 4.7 ns, and the propagation delay $t_p$ of a generic logic gate is 3.5 ns, estimate the maximum speed at which the rotator can work.

Problem 2. Analyse the output waveforms and deduce the binary codes $K[3..0]$ that generates the asynchronous circuit in Fig. 1 based on toggle flip-flops (T-FF).

- **Fig. 1** Diagram and function table of a of a T-FF and example asynchronous circuit.
- **Fig. 2** Output waveforms to be deduced from the circuit in in Fig. 1.
Problem 3. Microcontrollers.

We aim to implement a pulse-width modulation (PWM) generator that produces a 10 Hz fixed-frequency waveform, which duty cycle \( (DC = \frac{TON}{T\text{WAVE}}) \) could be modulated form 0% to 100%. See Fig. 3. The input will be the \( DC \) information encoded in 7-bit binary, so that \( DC = "0000000" \) represents the 0% modulation (all time ‘0’), and \( DC = "1100100" \) represents 100% (all time ‘1’). The start (ST) button toggles ON and OFF the output waveform and the ON_LED.

![PWM modulator diagram](image)

Fig. 3  a) Circuit to be designed, b) The ATmega8535 microcontroller, c) example of a rectangular waveform corresponding to a \( DC = 60\% \) (\( DC = "0101000" \)).

The general idea and the plan for the circuit may be to use a variable like \( Value_Q \) as a down counter that can be preloaded with the \( Value_Din = DC \) to count the \( T_{\text{ON}} \) time and then with the \( Value_Din = (100 - DC) \) to count the \( T_{\text{OFF}} \) time. In this way a PWM waveform is generated accordingly to the parameter \( DC \). For instance, if \( DC = 50\% \) a square waveform will be generated. The CLK for this down counter may be the external 1 kHz signal by means of generating an interrupt every 1 ms.

1. Hardware. Draw the schematic showing where all the signals may be connected along with the reset and quartz crystal oscillator of 4 MHz to run this application using an Atmel ATmega8535 microcontroller chip.

2. Software. Represent a hardware-software diagram for the FSM indicating which inputs to read and write and which will interrupt, the name and bit organisation of the RAM variables, etc. Explain how the C code will be organised and the purpose of each function.

3. Draw a state diagram for the FSM indicating transitions and outputs, and name the states: \( \text{Idle}, \text{Load}_{\text{TON}}, \text{DownCount}_{\text{TON}}, \text{Load}_{\text{TOFF}}, \text{DownCount}_{\text{TOFF}} \). Describe the tasks assigned to each state (\( \text{output\_logic} \)), and how the system will evolve from one state to the next (\( \text{state\_logic} \)).

4. Explain how to initialise some microcontroller pins to be digital inputs and some of them to be digital outputs (\( \text{init\_system} \)).

5. Draw the flow charts and the equivalent C code for the functions \( \text{read\_inputs}() \), \( \text{write\_outputs}() \) and \( \text{ISR}() \).

6. Draw the truth tables and their equivalent flow charts for the \( \text{state\_logic}() \) and \( \text{output\_logic}() \).