JAYPEE UNIVERSITY OF ENGINEERING & TECHNOLOGY, GUNA
DEPARMENT OF COMPUTER SCIENCE & ENGINEERING
Course: Computer Organization & Architecture Lab
Course Code: CS208/18B17CI474
B. Tech. (CSE IV/VI Sem.)

Experiment # 4

Aim: Design of logic circuits using mux, encoder and seven segment displays.

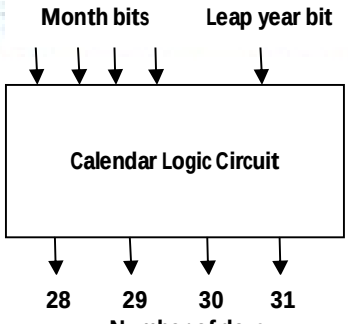
Exercise#1: Design full adder using **in-built block** of (i) 4:1 multiplexer with initial carry as input line and, (ii) 8:1 multiplexer.

Exercise#2: Design a logic circuit (using **in-built block** of priority encoder to identify whether the input octal number is even, odd or prime. Use one hex display to show input octal digit and a seven segment display at output which display **'E', 'O'** and **'P'** if input octal number is even, odd and prime respectively.

Design Steps:

- Find truth table with three inputs and three outputs.
- Derive four Boolean expressions for outputs using K-map.
- Implement the logic diagram as per derived Boolean expression and verify it.
- Add one hex display at input.
- Obtain truth table with three inputs and seven outputs for a seven segment logic circuit and derive seven Boolean expressions..
- Connect a seven segment at output as per the derived Boolean expressions.

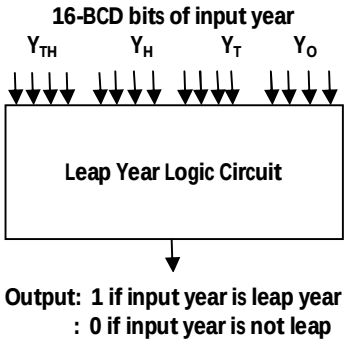
Exercise#3: Design calendar logic circuit to show the number of days in a given month. Display input month and leap year using two **hex displays** and number of days in outputs using **two seven segment displays** (two displays for each output).



Design Steps:

- Find truth table with five inputs and four outputs.
- Derive four Boolean expressions for outputs using K-map.
- Implement the logic diagram as per derived Boolean expression and verify it.
- Derive Boolean expressions for both the segments and make the connection in seven segments at output as per the obtained Boolean expressions.

Exercise#4: Design logic circuit to determine whether the input year is leap year or not. Use **four hex displays** to show four decimal digits of input year and **one hex display** to show output.



Example: input year 1986 will have 16-BCD bits as following:

Y_O: BCD bits for ones place digit 6 = 0110
Y_T: BCD bits for tens place digit 8 = 1000
Y_H: BCD bits for hundreds place digit 9 = 1001
Y_{TH}: BCD bits for thousands place digit 1 = 0001

Design concepts to derive Boolean expression for leap year:-

- All years ending with 00, 04, 08, 12, 16, 20, etc. are divisible by 4 and,
  - **a.** If tens digit is even (Y_{T0} = 0), and the ones digit is 0, 4, or 8.
  - **b.** If tens digit is odd (Y_{T0} = 1), and the ones digit is 2 or 6.
  - **c.** Digits with values of 10 to 15 will never be used.

  Using above three concepts, derived Boolean expression for D4 = $\overline{YT0}\,\overline{YO1}\,\overline{YO0}$ + YT0 YO1 $\overline{YO0}$.

- All years ending with digits 00 will be divisible by 100. It means lower eight bits (Yo and Y_T) will be always 0. Therefore, Boolean expression for the year divisible by 100 is D100 = $\overline{YT} \bullet \overline{Yo}$.

- All years which are divisible by 4 **(applied to the thousands and hundreds digits)** and 100 both will be divisible by 400. Therefore, Boolean expression for the year divisible by 400 is D400 = D4•D100.

- All years which are **divisible by 4 but not by 100** or **divisible by 400** are the leap years. Therefore, final Boolean expression to determine leap year = $D4 \bullet \overline{D100} + D400$.