# UNIVERSITY OF SWAZILAND

## FACULTY OF SCIENCE
## DEPARTMENT OF PHYSICS

**MAIN EXAMINATION 2005**

Title of the Paper: **DIGITAL ELECTRONICS**

Course Number: **P411**

Time Allowed: **Three Hours**.

Instructions:

1. Answer any five questions.
2. Each question carries 20 points

THIS PAPER HAS 7 PAGES, INCLUDING THIS PAGE

DO NOT OPEN THE PAPER
UNTIL PERMISSION HAS BEEN GIVEN BY THE INVIGILATOR.

**QUESTION 1:**

**a** Transform the Boolean function, $F = BE + \overline{B}D\overline{E}$, into:
(1) a K-map,
(2) a truth table: use hexadecimal numbers instead of binary code in the table, for example, $12_{hex}$ instead of $10010_{bin}$.
(Hint, expand this function into canonical form first) **10pts:**

**b** Using the tabulation method, simplify the following Boolean function F into either a sum of products or a product of sums (not both):
F(v, w, x, y, z)= $\Sigma$(2, 6, 9, D, 12, 16, 19, 1B, 1D, 1F)
(hex number in the brackets of the above function) **10pts:**

**QUESTION 2:**

**a** With the help of a K-map, obtain the simplified expressions in (1) SOP and (2) POS of the one of the following two Boolean Functions: (you are allowed to choose only one function and obtain the two expressions of the one you picked)

$G = \overline{ABC} + A\overline{B}D + \overline{AB}C\overline{D} + A\overline{B}\,\overline{D} + ABC$

$F = (\overline{A} + \overline{B} + D)(A + B + \overline{D})(A + \overline{B} + C + D)(\overline{A} + \overline{D})$ **10pts:**

**b** Implement the Boolean function, $F = (A + \overline{B})(CD + E)$, with only NAND gates and nothing but NAND gates. Complement inputs are available only at input terminals, nowhere else. The implement must have its function support. **10pts:**

## QUESTION 3:

**a** Implement the following function with a multiplexer of 3 bits select address (must have this component) and other elementary gates.
$F(v, w, x, y) = \Sigma(0, 1, 3, 5, 8, 9, F)$ **10pts:**
(hex number in the brackets of the above function)

**b** Fig. 3 are components for making a microprogrammed sequencer. In the ROM, there may be fields in a byte: jumping address, mux select address, and sequence output, C. Design a sequencer with only the components supplied, nothing else available. In all, draw a circuit diagram with the available components. **10pts:**

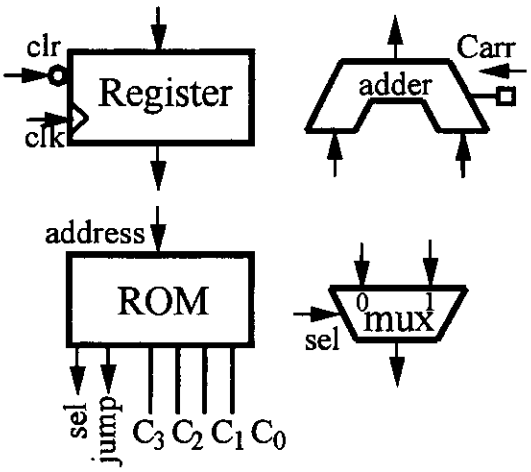


Fig. 3

**QUESTION 4:**

Suppose two 2-bit numbers, $B_1B_0$ and $A_1A_0$, are to be compared. The comparator will have three outputs: "B=A", "B>A", and "B<A". Assume that B and A are unsigned binary integers. Design a circuit to fit in a PLA. Simplification K-map is a must (hint, put the truth table directly into the K-map.) **20pts:**

**QUESTION 5:**

Design, with RS-ff's, a clocked sequencer to cycle repeatedly through the states: 0, 6, 2, 2, 0, 7. Obtain a logic circuit, a state table, a state diagram, and ff input functions. (Hint: need a hidden unit) **20pts:**

**QUESTION 6;**

Design a sequential machine, with no restriction on the use of any logic components. Its ASM diagram is shown in Fig. 6 below. Obtain a state transition table, and a circuit diagram plus the support of the logic equations. Two D-ff's are proper to use. **20pts:**

Fig. 6, ASM diagram