Topic 7
Exercise
Min/Max finder
Design of Data path unit (DPU) and identifying control signals required to control the DPU. Control unit design using one-hot method as well as microprogrammed control unit method.

Example: Sixteen 4-bit unsigned numbers are stored in a 16x4 memory. You need to search and find the largest number and the smallest number. Assume the availability of two 4-bit comparators. Think of an appropriate data unit, identify control signals, and design a control unit to go with it. Repeat the design for the case of a single comparator. Remember that, unlike in software, you try to perform in hardware as many operations as possible in parallel. This is to achieve speed. For example, you can increment iteration counter while performing the iteration and simultaneously check if it is time to exit the iteration loop. How many clocks do you think you need to complete the above job once you are given a START command?
Items in DPU for the case of two comparators: There must be a memory (16x4) and two comparators as per the problem statement. In addition we need ____________ data register(s) to hold the largest and smallest numbers found in the iterative search.

Algorithm: After the START is available, load the first (zero-th) number (in the memory) in both the MAX and MIN registers. Enter an iterative loop and compare subsequent numbers (in the memory) with both the running largest and smallest numbers found so far, updating the MAX and MIN registers appropriately. When all iterations are done (or about to be done!!!???) go to a done state.

Draw a flow chart and a state diagram.
Data Unit for the case of two comparators:

Control Unit:
Data Unit for the case of two comparators:

Control Unit:

Counter
FLOW CHART (not a State Diagram!)

START?

NO

YES

i <- 0

MAX <- M(i)

MIN <- M(i)

i <- i + 1

compare M(i) with MAX

NO

M(i) > MAX?

YES

MAX <- M(i)

compare M(i) with MIN

NO

M(i) < MIN?

YES

MIN <- M(i)

i <- i + 1

compare i with max count

Is i = max count?

NO

YES

DONE

Back to INITIAL
FLOW CHART (not a State Diagram!)

- **START?**
  - **NO**
    - **i < 0**
    - **NO**
      - **MAX <= M(i)**
      - **MIN <= M(i)**
      - **i <= i + 1**
    - **YES**
      - **M(i) > MAX?**
        - **YES**
          - **MAX <= M(i)**
        - **NO**
          - **M(i) < MIN?**
            - **YES**
              - **MIN <= M(i)**
            - **NO**
  - **YES**
    - **i <= i + 1**
    - **compare i with max count**
    - **Is i = max count?**
      - **NO**
        - **DONE**
      - **YES**
        - **Back to INITIAL**

**INITIAL**
FLOW CHART (not a State Diagram!)

INITIAL

NO
START?
YES

i <- 0

MAX <- M(i)

MIN <- M(i)

i <- i + 1

compare M(i) with MAX

NO

M(i) > MAX?
YES
MAX <- M(i)

compare M(i) with MIN

NO

M(i) < MIN?
YES
MIN <- M(i)

i <- i + 1

compare i with max count

NO

i = max count?
YES
DONE

Back to INITIAL
Single comparator case, Mealy machine:
Write the state transitional conditions. List outputs to be generated or control actions to be performed in each state.
Moore m/c

Mealy m/c