Indian Institute of Information Technology, Sri City, Chittoor

Name of the Exam: Cyber-Physical Embedded Systems Duration: 4 hours (1100-1500) Max. Marks: 75

Roll No.:____________________________________Room No.: _______________Seat No.: __________

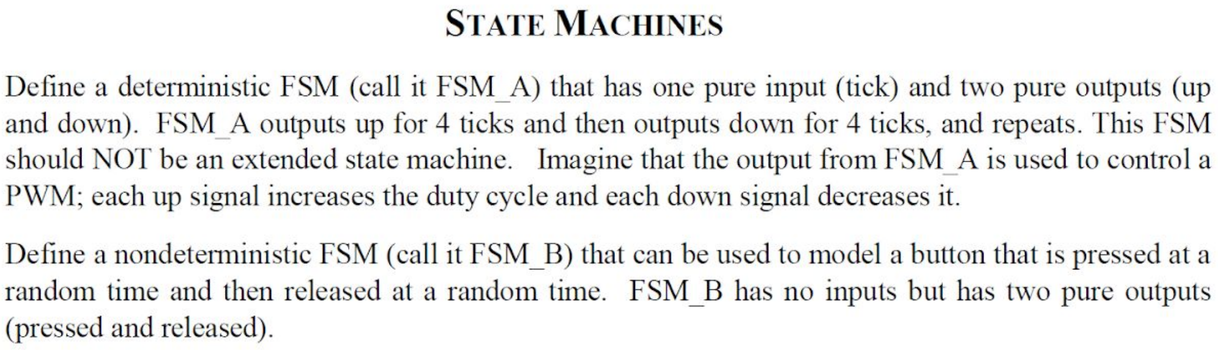
Name: ______________________________________Invigilator’s Signature: ________________________

Instructions:

1. Students are required to write the answers in A4 sheets. Students are required to clearly write their roll number, name in capital letters on top of every page of the answer sheets. Please mention your question paper number in your answer sheet (**this is question paper number 1**).
2. At the end of the exam, students are expected to submit the scanned copy of the answer sheets to balaji.r@iiits.in as per the indicated closing time.
3. Answer all questions.
4. It is a closed book exam.
5. If there are some values not given in this question paper, kindly assume standard values and state your assumption in the answer sheet. If you don’t know the standard values use variable names and derive solutions with unknown variables.
6. **Students have to clearly give the answers in a step by step mode. If the student answers the questions directly giving an answer then the student may get a small set fraction of the total mark.**

---

1. *Finite State Machines:* Answer the following two sub questions. Each sub question carries 15 marks. So in total 30 marks. For both the below questions draw the finite state machine and present the mathematical model for it. If they are any behavior and/or parameters not defined assume appropriately. Based on the system definition choose if you need a deterministic or non-deterministic state machine. Along with this question paper, there is a handout given for pulse width modulation.

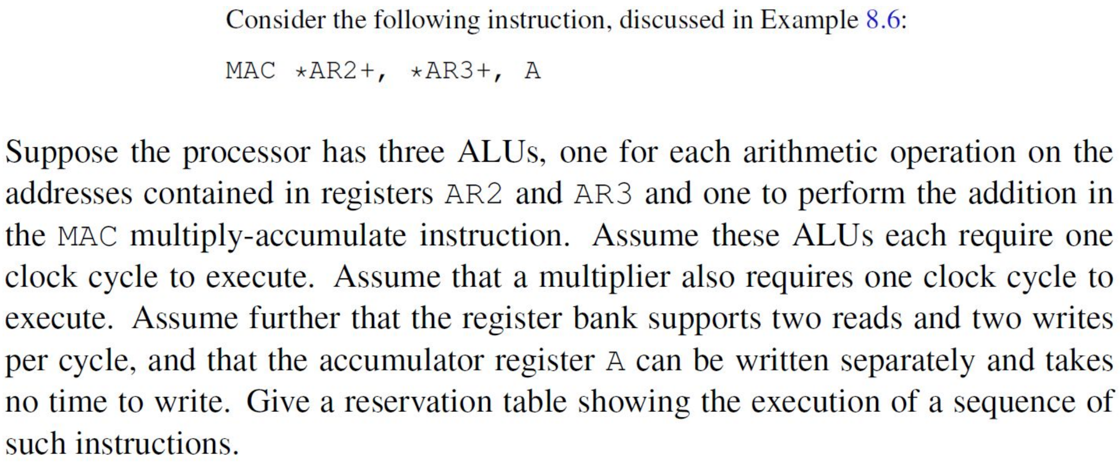


STATE MACHINES

Define a deterministic FSM (call it FSM_A) that has one pure input (tick) and two pure outputs (up and down). FSM_A outputs up for 4 ticks and then outputs down for 4 ticks, and repeats. This FSM should NOT be an extended state machine. Imagine that the output from FSM_A is used to control a PWM; each up signal increases the duty cycle and each down signal decreases it.

Define a nondeterministic FSM (call it FSM_B) that can be used to model a button that is pressed at a random time and then released at a random time. FSM_B has no inputs but has two pure outputs (pressed and released).

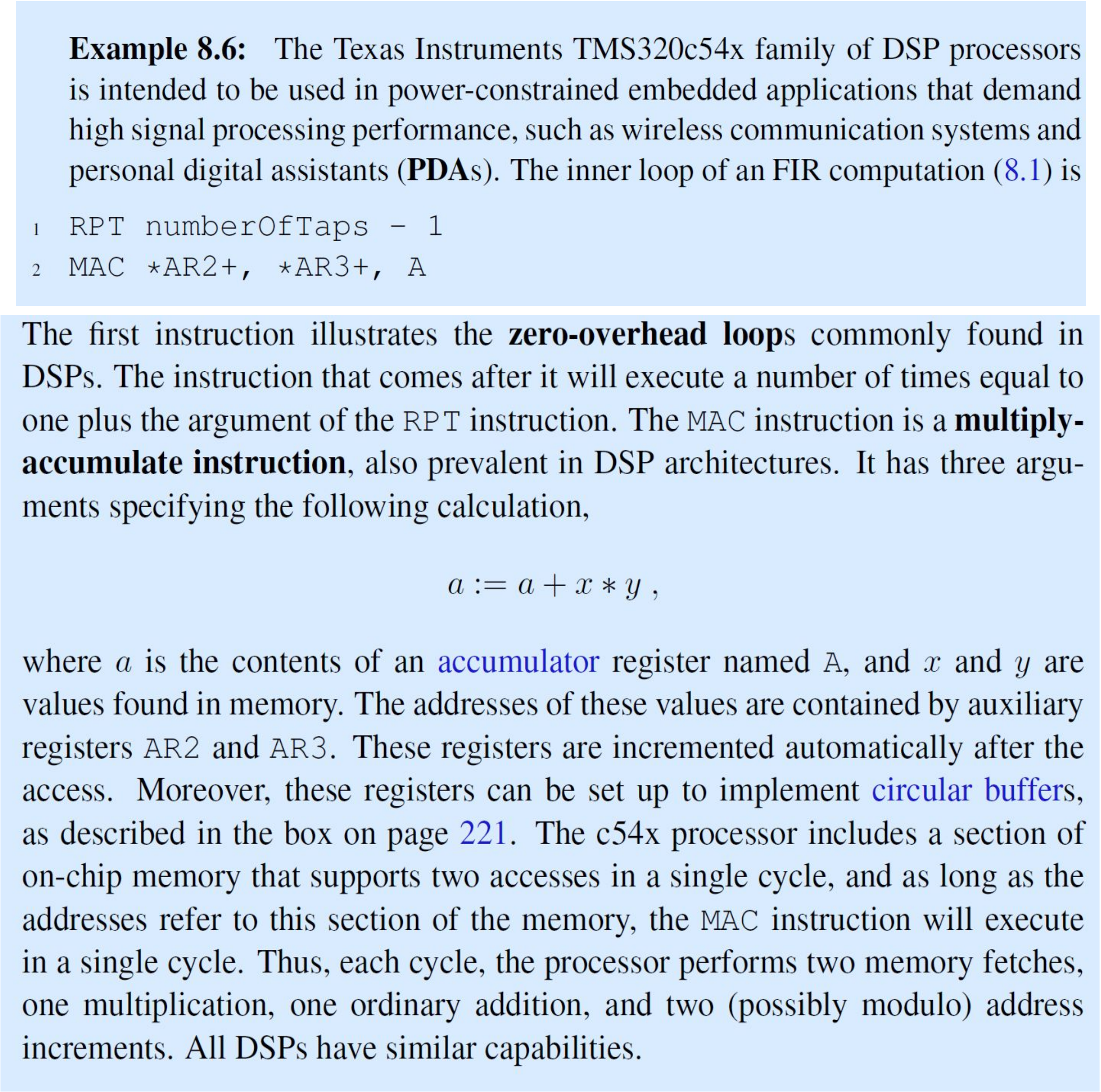
2. *Embedded Processors:* Answer the following question. This question carries 15 marks. The question is immediately given below this text. As a reference to this question the Example 8.6 is given from the textbook as a shadowed box.



Consider the following instruction, discussed in Example 8.6:

```
MAC *AR2+, *AR3+, A
```

Suppose the processor has three ALUs, one for each arithmetic operation on the addresses contained in registers AR2 and AR3 and one to perform the addition in the MAC multiply-accumulate instruction. Assume these ALUs each require one clock cycle to execute. Assume that a multiplier also requires one clock cycle to execute. Assume further that the register bank supports two reads and two writes per cycle, and that the accumulator register A can be written separately and takes no time to write. Give a reservation table showing the execution of a sequence of such instructions.



Example 8.6: The Texas Instruments TMS320c54x family of DSP processors is intended to be used in power-constrained embedded applications that demand high signal processing performance, such as wireless communication systems and personal digital assistants (**PDA**s). The inner loop of an FIR computation (8.1) is

```
RPT numberOfTaps - 1
MAC *AR2+, *AR3+, A
```

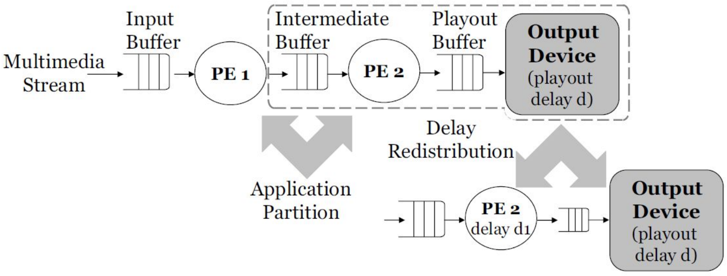
The first instruction illustrates the **zero-overhead loop**s commonly found in DSPs. The instruction that comes after it will execute a number of times equal to one plus the argument of the RPT instruction. The MAC instruction is a **multiply-accumulate instruction**, also prevalent in DSP architectures. It has three arguments specifying the following calculation,

$$a := a + x * y ,$$

where a is the contents of an accumulator register named A, and x and y are values found in memory. The addresses of these values are contained by auxiliary registers AR2 and AR3. These registers are incremented automatically after the access. Moreover, these registers can be set up to implement circular buffers, as described in the box on page 221. The c54x processor includes a section of on-chip memory that supports two accesses in a single cycle, and as long as the addresses refer to this section of the memory, the MAC instruction will execute in a single cycle. Thus, each cycle, the processor performs two memory fetches, one multiplication, one ordinary addition, and two (possibly modulo) address increments. All DSPs have similar capabilities.

3. *Discrete /Continuous Dynamics: (30 marks)* For the below specification given:
   a. Construct the Actor model for design and analysis of the below problem statement *(20 marks)*,
   b. Write the formal model for the actor model you have constructed. *(10 marks)*.

Consider our system model shown in Figure, and assume a streaming application that is partitioned into tasks and mapped to the Processing Elements (PEs) of the system. A constant bit rate input stream arrives at the input buffer, PE 1 partially processes it and writes it to the intermediate buffer. PE 2 reads items from the intermediate buffer, completely processes the stream, and writes it to the playout buffer. Finally, the display device consumes items from the playout buffer at a constant rate. The display device reads items from the playout buffer after an initial delay *d*. At time instance zero, PE1 starts to process items from the input buffer. After a delay *d1*, PE2 processes items from the intermediate buffer. The display device displays 30 frames per second.


Input Buffer
Multimedia Stream
PE 1
Intermediate Buffer
PE 2
Playout Buffer
Output Device (playout delay d)
Delay Redistribution
Application Partition
PE 2 delay d1
Output Device (playout delay d)