Question Set : 2

Name: KRISHNA KUMAR DEY

Roll No: S20160010046

Q1 a) Pipelining is one way of improving the overall processing performance of a processor. This approach allows the simultaneous execution of several instruction. It exploits parallelism at instruction level by overlapping the execution process of instructions. It is analogous to an assembly line where workers perform a specific task & pass the partially completed product to the next worker.

The throughput H, also called bandwidth, of a pipeline is defined as number of input tasks it can process per unit of time.

Over an equivalent sequential processor without pipeline, ~~increases~~ ~~the~~ the pipelined processor has more throughput

But pipelining doesnot reduce the execution time of an individual instruction. Moreover, it slightly increases the execution time of each instruction due to overhead in the pipeline control. This leads to increase in latency.

Scanned with CamScanner

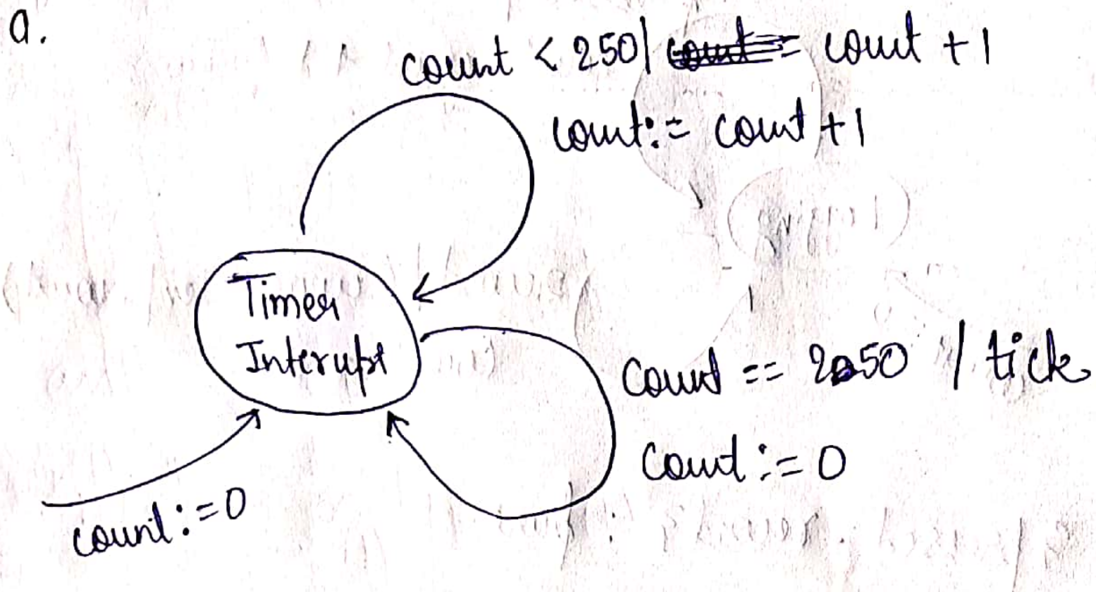
Q1. b

As the processor is not handling data hazard which means that the next instructuction is Independent of the previous one. This can lead to the conclusion that if the sequential instructions in the program are independent of each other only then the data hazard doesn't occur and the processor doesn't need to take care for that or the dependant instruction comes only when the instruction it depends on executes and write its result to the register.

Scanned with CamScanner

Set - 2

Q2 a.



Input = { }

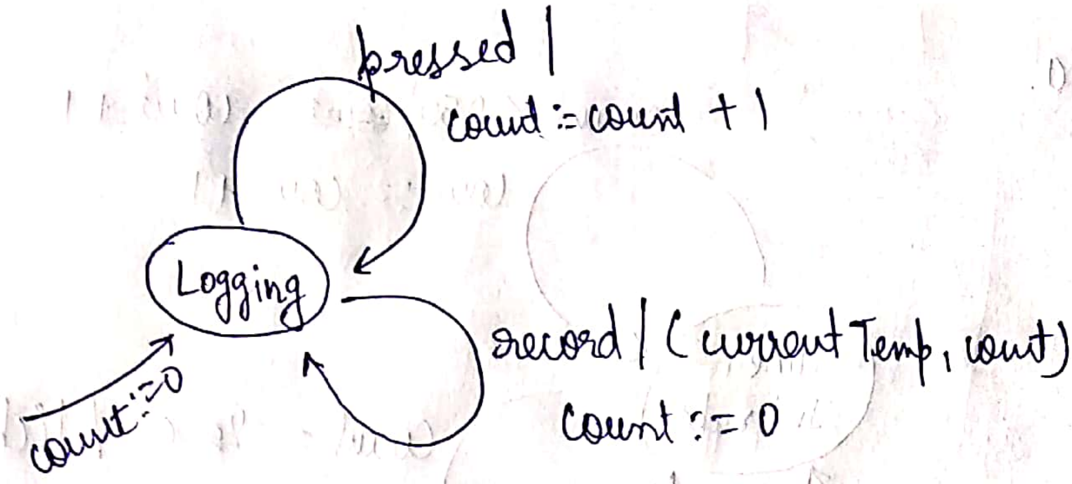
Output : tick : pure

Variable : ~~count~~ count : {0, 1, .. 249}.

Since there is input required in the above FSM, we can say that this is an extended state FSM. For the variable 'count' there are 250 possible values or states in the simple FSM. Output of this FSM is a tick signal. This tick signal is pure as Vp of tick ∈ {present} ∪ {absent}. Whenever the ~~tick~~ count reaches 250 then is a tick & count reset to 0. This cycle continues indefinetely.

Scanned with CamScanner

Q2 b.



Input : {pressed , record} : pure

Output : {current Temp , Count}

Variable : Count $\in \mathbb{Z}_+ \cup \{0\}$

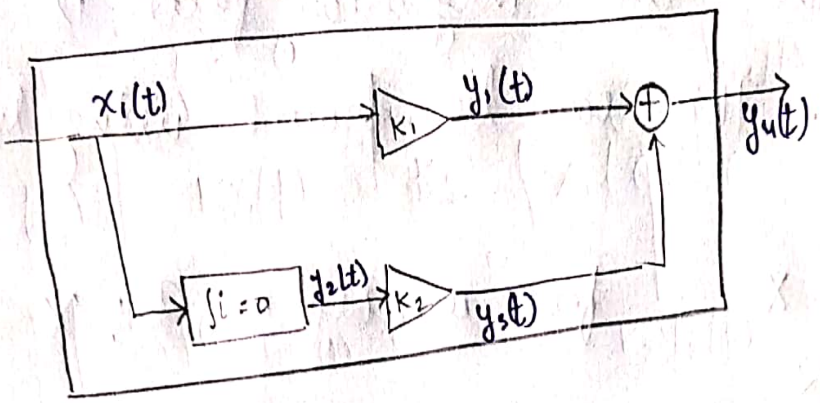
In the above FSM, the record input will have a different FSM which counts till 30 sec and gives out a signal {present} to record signal to capture the log count and the current temperature at that point. The count variable could have any (+)ve integer along with 0.

The outputs are currentTemp and the count of the number of times the button was pressed. After record the count is set to 0.

The extended state machine can have ∞ no of states which is equal to no of times button could be pressed.

Set 2

Q3. a



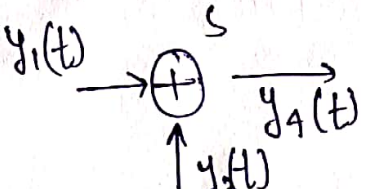
$x_1(t) : \mathbb{R} \to \mathbb{R}$

$\forall\ t \in \mathbb{R},\ y_1(t) = K_1\, x_1(t)$

$\forall\ t \in \mathbb{R},\ y_2(t) = \cancel{i}^{\,0} + \int_0^t x_1(\tau)\,d\tau$

$\forall\ t \in \mathbb{R},\ y_3(t) = K_2\, y_2(t)$

$= K_2 \int_0^t x_1(\tau)\,d\tau$



$S : (\mathbb{R} \to \mathbb{R})^2 \longrightarrow (\mathbb{R} \to \mathbb{R})$

$\forall\ t \in \mathbb{R},\ y_4(t) = y_1(t) + y_3(t)$

$= K_1 x_1(t) + K_2 \int_0^t x_1(\tau)\,d\tau$

Scanned with CamScanner

Q3 b.

As we know that an integration factor is strictly causal ~~casual~~. This can be proved using below property.

$\forall\ x_1, x_2 \in X$ for a system $S : X \to Y$ and $\tau \in \mathbb{R}$

$$x_1|_{t<\tau} = x_2|_{t<\tau}$$

$$\Rightarrow \quad S(x_1)|_{t\leq\tau} = S(x_2)|_{t\leq\tau}$$

Also we know that adder is a causal

$\therefore$ model is causal as it is comprising of integrator & adder.

Scanned with CamScanner