1.

A.

A state transition diagram consists of named states, represented by boxes or ovals. Arcs join states together, the arcs are labelled with the condition that causes the transition and the (optional) actions that occur during the transition. State transition diagrams are useful for describing the operation of synchronous digital logic systems. They are also used in UML modelling to show the states of a system of any kind.

B.

[hand drawn diagram to follow]

C.

Inputs:

on button (optional, could just go to start state)
water level sensor
water temperature sensor
timer expiry
Sufficient bits to represent all the states

Outputs:

Hot water valve
Cold water valve
Drain valve
Heater on
Timer start
Timer value
Sufficient bits to represent all the states

D.

[sensible suggestions sought, no single answer]

If an emergency stop were to be used then it should perhaps cause both water valves to be closed, the heater to be turned off and the drain valve to be opened. This would require a new state, with a transition from each of the other states.

2.

A.

The MIPS processor provides the “Jump and Link” instruction to jump to a program address while storing the current program counter + 4 value into a register. There is also a “Jump Register” instruction which transfers execution to the address given in a register, allowing procedure returns.

B.

$a0 - $a3 are used for the storage of arguments, passed into the procedure
$v0 - $v1 are used to return values from the procedure to the callee

$t0 - $t9 are temporary storage that can be used and over written by the procedure as required
$s0 - $7 can be used by the procedure but MUST be saved and restored before the procedure returns

C.

A stack is a first-in-last-out data structure which can be implemented by an array and a pointer. By convention in the MIPS processor, one of the general purpose registers (known as the $sp, the stack pointer) is used as the pointer into an area of memory representing the stack. To put information onto the stack it is stored at the location pointed to by $sp and then $sp is decremented (by convention, MIPS stacks grow downwards). To pull information off the stack it is read as required and the stack pointer is incremented. Thus additional arguments to procedures can be placed on the stack before the procedure is called and the procedure can retrieve them as required. A second register, known as the frame pointer, $fp, keeps track of where the data stored "locally" by the procedure on the stack is. Thus local data stored on the stack can be accessed relative to the frame pointer, which remains fixed for the duration of the procedure call. The stack pointer can move during the procedure call, if for example the procedure places more data onto the stack.

D.

The MIPS processor does not provide explicit PUSH and POP instructions for stack use as this would not fit cleanly with the pipelining approach, the PUSH instruction for example requires both that a word is stored in memory at a location pointed to by a register and that register is decremented in the same instruction. This creates a structural hazard for the pipeline.

3.

A.

A minimal sum-of-products form of a Boolean equation is a set of AND terms, all of which are OR'd together, i.e. the equation can be written without brackets. A minimal sum-of-products form is one in which the equation is described by the smallest possible number of product terms. This is a useful representation as it describes the smallest amount of digital logic that is required to implement the function

B..

i. A minterm is one of the AND terms in the sum-of-products form of a boolean equation.

ii. A prime implicant is a group of adjacent 1's on a karnaugh map

iii An non-essential prime implicant is a prime implicant that includes at least one 1 that is covered by some other non-essential prime implicant.

C.

[hand drawn diagrams to follow]

D.

The Karnaugh map can only easily be used for a small number of variables, 4 for a single map, 5 or 6 for multiple maps. Q-M has no theoretical limit on the number of variables, however in some cases the method suffers from exponential complexity in the number of variables. QM is used to minimise the logic functions described by designers in a high level design language so that they can be implemented in the smallest number of logic gates in hardware.

4.

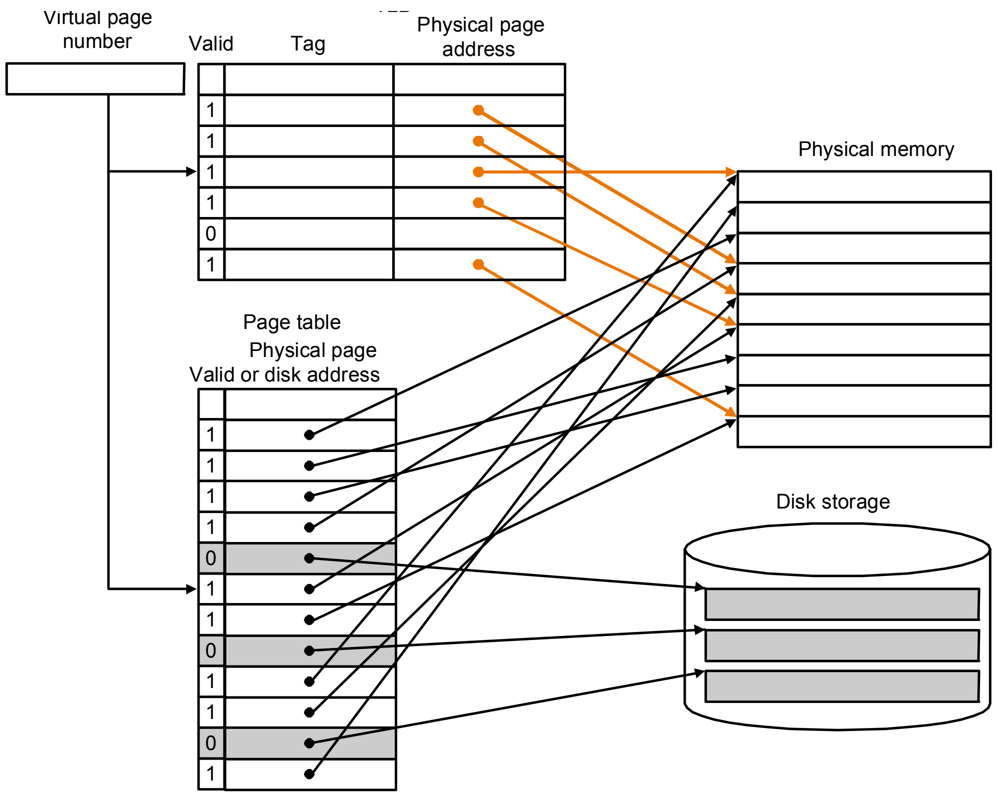
A.

A virtual memory system allows more than one process to exist on the processor and for each process to have its own (possibly protected) memory space and for that memory space to be larger than the physically available memory. It differs from cache memory in that cache memory operates on a single process, over the space a few thousand instructions while that process is active on the processor. Virtual memory operates and the level of the entire process. They are complementary techniques which can both improve processor utilisation and throughput.

B.

A translation lookaside buffer holds recently used page table mappings of physical addresses for a virtual memory system. It removes the need to access the page table for some memory references and hence can speed up processing as it reduces the number of memory accesses required to locate data. Its operation is shown in the diagram below:

[similar diagram or narrative description also acceptable]



C.

- Hardware generates an exception
- “offending” address stored in EPC
- Current process suspended & exception handler is invoked
- If instruction miss, need page with EPC address

- If data miss, must read (& decode) instruction at EPC to find required address
- Start disk request for reading of page
  - Hand control to scheduler
  - Disk read completes, original process now runnable

D.

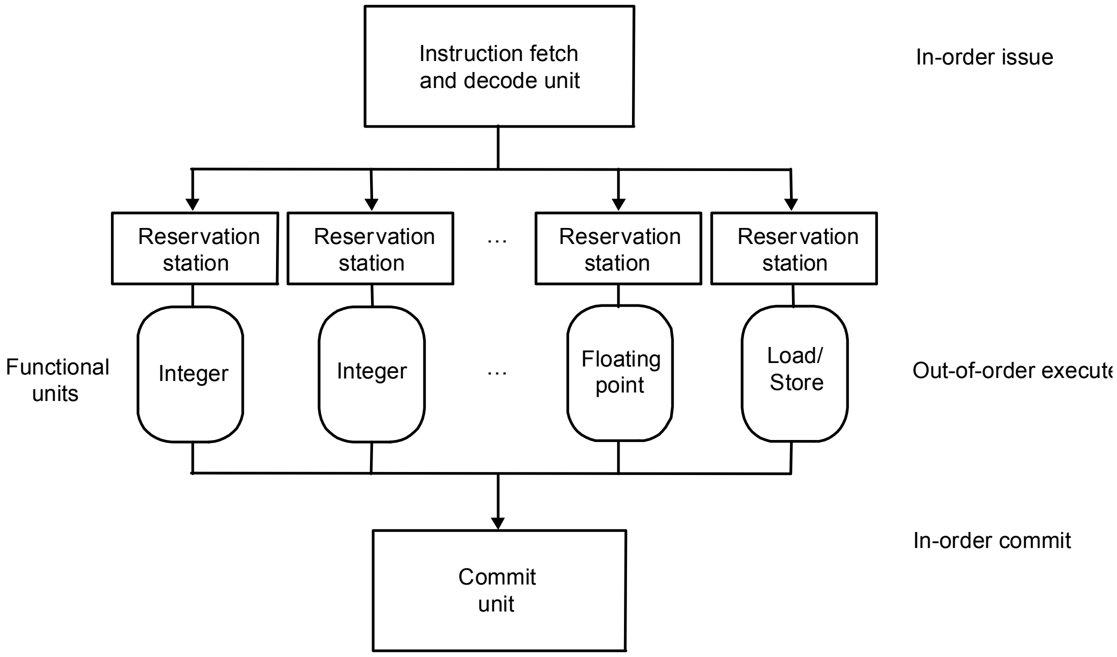
We could replace the page that was least recently used, requiring use to keep a record of which pages were used when, or we could replace the page that is oldest, i.e. has been in the page table for the longest time.

5.

A.

A superscalar processor duplicates the entire pipelined datapath, and thus multiple instructions (1 per pipelined datapath) can be launched each clock cycle. This can only be done if the instructions are independent, or if they are of different types, for example 1 floating point and 1 integer or logical instruction each clock cycle. The limitations arise from the additional hardware requirements, both in duplicating the datapath and in the additional control logic required, and the greater the number of duplicated datapaths the harder it becomes to find totally independent instructions to keep each pipeline full.

B.



When an instruction cannot be added to the pipeline because of a hazard, a dynamic scheduling processor would attempt to find an instruction which can be executed and allocate to one of the execution units. The reservation station holds the operation and the operands and executes the operation when all are available. The commit unit writes the registers and memory when all previous instructions have been committed.

C.

Caching is a viable approach to improving processor performance because over a significant proportion of the execution time most programs operate on only a small amount of data, which is located within a small area of the address space. Similarly, over a significant proportion of execution time most of the code executed is restricted to a small area of the program, large scale jumps are relatively infrequent.

D.

[Anything highlighting some of the main features from following list]

- SRAM on the same die as the CPU
  - 5ns / Relatively large area of silicon per bit / v. high cost
- Separate SRAM between the CPU and the bus
  - 20ns / additional chips / high cost
- DRAM, attached to the system bus
  - 60ns / less silicon per bit (higher density) / medium cost
- Magnetic disk etc. (mass storage)
  - 1s / extremely high density / very low cost

6.

A.

Control signals are generated at the instruction decode stage of the pipeline, by control logic. These signals determine the operation of, for example, the ALU and various multiplexors; however these signals are only needed at further stages in the pipeline. The ALU signals are needed in the instruction execution phase for example, and hence buffering is required between the instruction decode phase and the instruction execution phase to ensure that the signals arrive at the correct time.

B.

A hazard is a situation in which an instruction cannot be overlapped with another instruction already in the pipeline for some reason. A non-pipelined processor does not exhibit this problem because only instruction executes at a time and hence is completed before the next instruction starts.

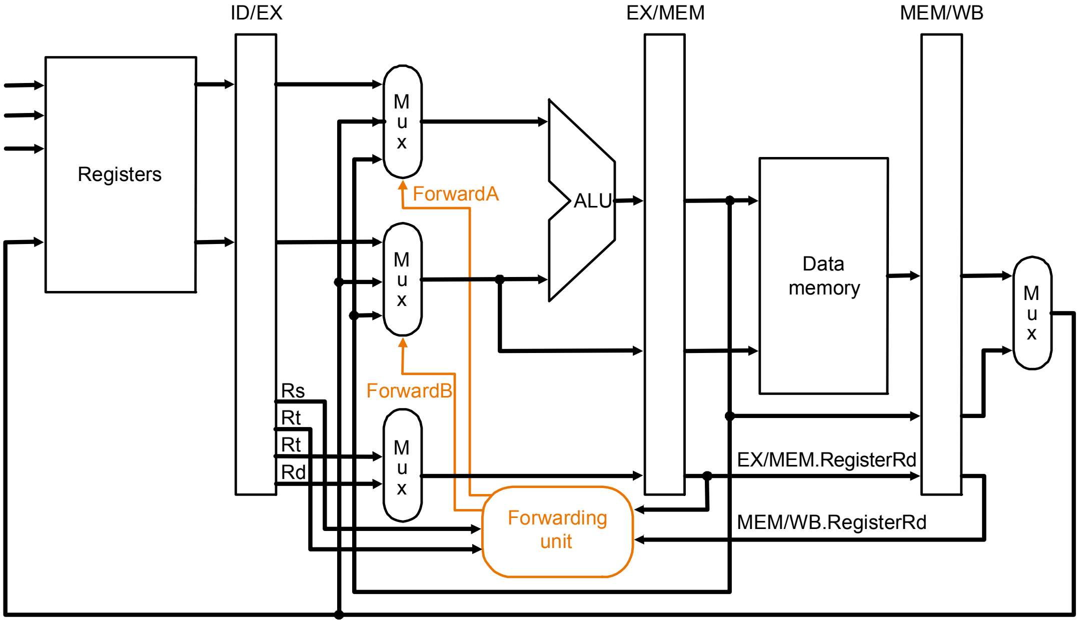
C.

A data hazard exists where one or more of the operands is the result of an instruction already in the pipeline and hence not yet available.

One approach is to "stall" the pipeline, and to stop issuing instructions to the pipeline until all of the operands are available.

A second approach is to use "register forwarding", as shown in the diagram below.

[anything approximating that below, highlighting the addition of the forwarding unit and the links from buffer registers **back** to the execution stage]



b. With forwarding

D.

A 2 bit branch prediction scheme uses two bits to record whether a branch at a particular address was taken or not. Only if a prediction is wrong twice is the assumption changed. This improves on the performance of a single bit scheme because the single bit will mis-predict twice on each loop – once as a leftover from the previous time this loop was exited, and once at the end of the loop. The two bit scheme will not be "fooled" by the previous exit from the loop.