Computer Design (2004) Paper 6 Question 2 (SWM)

Sketch Answer

(a) Control hazards arise when the program counter moves from its sequential path, e.g. due to a branch instruction. Instructions that have been speculatively fetched and inserted into the pipeline have to be dealt with, typically by removing them from the pipeline.

(b) Data hazards arise when new data is produced by one stage in the pipeline but has not yet been written back to the register file. In the 6 stage pipeline, the execute and memory access stages produce new data values. Data hazards could be resolved by stalling the pipeline when a data hazard arises and then waiting until the data is written back to the register file.

(c) Stalling the pipeline in order to resolve data hazards is wasteful of processor cycles. Consequently, extra buses are added to forward recent results back to earlier stages in the pipeline. These so called feed forward paths reduces the need for stalls. In the 6 stage pipeline, the execute and memory access stages can produce new results. The shift and execute stages take new register values as input. Consequently, we would need feed forward paths from the outputs of the execute and memory access stages back to the inputs of the shift and execute stages.

(d) As described in part b, a branch instruction causes a control hazard. These hazards may be resolved by removing erroneously fetched instructions from the pipeline. These purged instructions pass through the pipeline and are referred to as bubbles in the system. In the case of the 6 stage ARM pipeline, the branch instruction is likely to be executed at the Execute stage at which point three instructions in the earlier three stages will have been fetched in error and need to be purged from the pipeline.