



Data path and control consideration

Datapath and control considerations: Consider the three-bus structure suitable for pipelined execution with a slight modification to support a 4-stage pipeline as shown in figure 3.18.

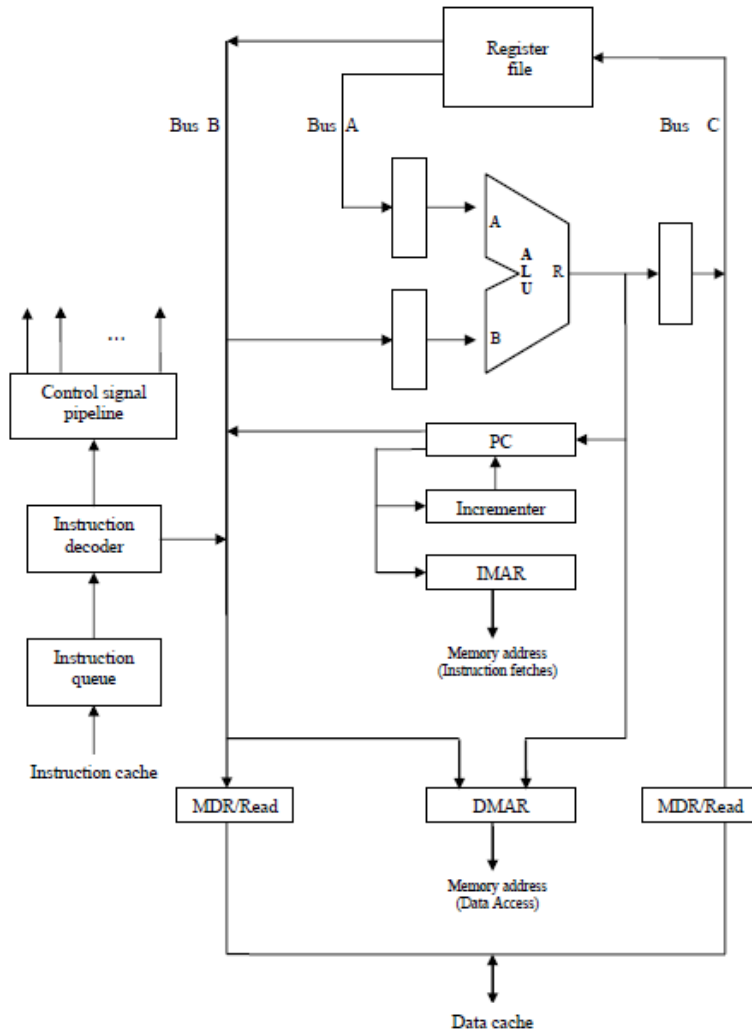


Figure 3.18 Datapath modified for pipelined execution with interstage buffers at the input and output of the ALU.

Several important changes are

1. There are separate instruction and data caches that use separate address and data connections to the processor. This requires two versions of the MAR register, IMAR for accessing the instruction cache and DMAR for accessing the data cache.
2. The PC is connected directly to the IMAR, so that the contents of the PC can be transferred to IMAR at the same time that an independent ALU operation is taking place.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

3. The data address in DMAR can be obtained directly from the register file or from the ALU to support the register indirect and indexed addressing modes.
4. Separate MDR registers are provided for read and write operations. Data can be transferred directly between these registers and the register file during load and store operations without the need to pass through the ALU.
5. Buffer registers have been introduced at the inputs and output of the ALU. These are registers SRC1, SRC2, and RSLT. Forwarding connections may be added if desired.
6. The instruction register has been replaced with an instruction queue, which is loaded from the instruction cache.
7. The output of the instruction decoder is connected to the control signal pipeline. This pipeline holds the control signals in buffers B2 and B3 in Figure 3.3.

The following operations can be performed independently in the processor of Figure 3.18:

- Reading an instruction from the instruction cache
- Incrementing the PC
- Decoding an instruction
- Reading from or writing into the data cache
- Reading the contents of up to two registers from the register file
- Writing into one register in the register file
- Performing an ALU operation

The processor execution time T , of a program that has a dynamic instruction count N is given by

$$T = \frac{N \times S}{R}$$

Where S is the average number of clock cycles it takes to fetch and execute one instruction and R is the clock rate. This simple model assumes that instructions are executed one after the other, with no overlap. A useful performance indicator is the instruction throughput, which is the number of instructions executed per second. For sequential execution, the throughput, P_s is given by

$$P_s = R/S$$

In general, an n -stage pipeline has the potential to increase throughput n times. Thus, it would appear that the higher the value of n , the larger the performances gain. Any time a pipeline is stalled, the instruction throughput is reduced. Hence, the performance of a pipeline is highly influenced by factors such as branch and cache miss penalties.