B multiple Bus organization

multiple bus organization in computer architecture is a design that allows multiple device to work simultaneously. The reduces the time spend waiting and improves the computer's speed. The main advantage of multiple bus organization is the reduction in the number of cycles required for execution. In a multiple bus structure one bus is used to fetch instructions and the other is used to fetch data. The same bus is shared by three units: memory, processor and I/O units.

- Better connectivity
- An increase in the size of the registers.

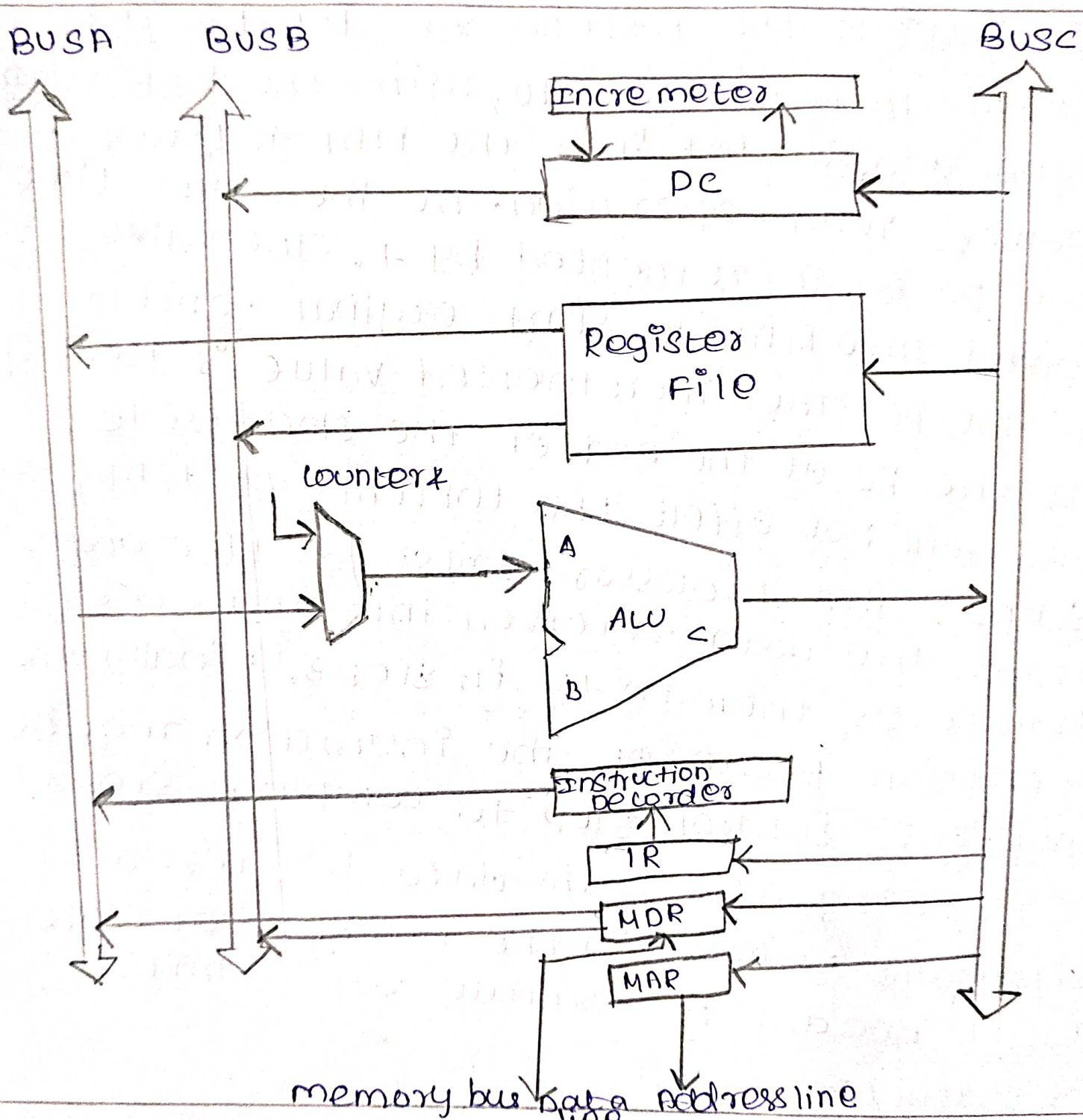
There are three types of bus lines:

Data bus, address bus, and control bus.

Three bus organization of data path:

In single bus organization only one data item can be transferred over the bus in a

clock cycle. To reduce the number of steps To reduce the number of steps needed most commercial processors provide multiple internal paths that enable several transfers to take place in parallel. All general purpose registers are combined into a single block called the register file. The register file have three ports. There are two outputs, allowing the contents of two different register file to be accessed simultaneously and have their contents placed on buses A and B. The third port allows the data on bus C to be loaded into a third register during the same clock cycle. Buses A and B are used to transfer the source operands to the A and B input of the ALU, where an arithmetic or logic operation may be performed. The result is transferred to the destination over bus C. If needed the ALU may simply pass one of its two input operands unmodified to bus C. The ALU control signals for such an operation R = A or R = B. A second feature incre - ment unit, which is used to increment the PC by 4. It can be used to increment other addresses such as the memory address in Load multiple and Store Multiple instruction.


BUSA
BUSB
BUSC
Increment
PC
Register File
Counter
A
ALU
C
B
Instruction Decoder
IR
MDR
MAR
memory bus
Data line
address line

Action : 1) PCout, R=B, MARin, Read, Inc PC
2) WFMC
3) MDRoutB, R=B, IRin
4) R1out, R2outB, SelectA, Add, Rbin, End.

In Step 1: the contents of the PC are passed through the ALU, using the R=B control signal and loaded into the MAR to start a memory read operation. At the same time the PC is incremented by 4. The value loaded into MAR to start orginal content of the PC. The incremented value is loaded into the PC at the end of the clock cycle and will not affect the contents of MAR.
Step 2: The processor waits for MFC and loads the data received into MDR, then transferes them to IR in step 3. Finally the Execution phase of the instruction requires only one control step to complete step 4.
By providing more paths data transfer a significant reduction in the number of clock cycles needed to execute an instruction is achived.