Register transfer:

For each register **Ri**, there are two control signals:

Ri_{in} used to load the data on the bus into the register.
Ri_{out} used to place the register's contents on the bus.

$R1 \rightarrow R4$ will be
$R1_{out,}$ $R4_{in}$

EX: What is the sequence of operation to the following instruction:

1. **MOVE R1, R2**
   1. $R1_{out}$, MDR_{in}
   2. MDR_{out}, $R2_{in}$

2. **MOVE (R1), R2**
   1. $R1_{out}$, MAR_{in}, Read
   2. WMFC (Wait to Memory Function Complete)
   3. MDR_{out}, $R2_{in}$

3. **MOVE R1, (R2)**
   1. $R2_{out}$, MAR_{in}, Read
   2. WMFC (Wait to Memory Function Complete)
   3. $R1_{out}$, MDR_{in}, Write
   4. WMFC (Wait to Memory Function Complete)

4. **MOVE (R1), (R2)**
   1. $R1_{out}$, MAR_{in}, Read
   2. WMFC (Wait to Memory Function Complete)
   3. $R2_{out}$, MAR_{in}, Write
   4. WMFC (Wait to Memory Function Complete)

To perform arithmetic/logic operation:

. The ALU is a combinational circuit that has no internal storage.

. ALU gets the two operands from MUX and bus. The result is temporarily stored in register Z.

EX: What is the sequence of operation to the following instruction:

1. ADD R1, R2

1. $R1_{out}$, Y_{in}
2. $R2_{out}$, Select Y, Add, Z_{in}
3. Z_{out}, $R2_{in}$

2. ADD (R1), R2

1. $R1_{out}$, MAR_{in}, Read
2. WMFC, $R2_{out}$, Y_{in}
3. MDR_{out}, Select Y, Add, Z_{in}
4. Z_{out}, $R2_{in}$

3. ADD R1, (R2)

1. $R2_{out}$, MAR_{in}, Read
2. WMFC, $R1_{out}$, Y_{in}
3. MDR_{out}, Select Y, Add, Z_{in}
4. Z_{out}, MDR_{in}, Write
5. WMFC

4. ADD (R1), (R2)

1. $R1_{out}$, MAR_{in}, Read
2. WMFC
3. MDR_{out}, Y_{in}
4. $R2_{out}$, MAR_{in}, Read
5. WMFC
6. MDR_{out}, Select Y, Add, Z_{in}
7. Z_{out}, MDR_{in}, Write
8. WMFC

Execution of Branch instructions:

A branch instruction replaces the content of PC with the:

branch target address(PC_{new})= offset **X** + PC_{old}
offset **X** (given in the branch instruction).

The required control steps to fetch and execute unconditional branch as follows:

1 PC_{out}, MAR_{in}, Read, Select 4, Add, Z_{in}
2 Z_{out}, PC_{in}, Y_{in}, WMFC
3 MDR_{out}, IR_{in}
4 offset X_{out}, Select y, Add, Z_{in}
5 Z_{out}, PC_{in}, End.

EX:
The required control steps to fetch and execute **a conditional branch** (**Br<0**) is illustrated bellow:

Note: N=0 (positive)
N=1 (negative)

1 PC_{out}, MAR_{in}, Read, Select 4, Add, Z_{in}
2 Z_{out}, PC_{in}, Y_{in}, WMFC
3 MDR_{out}, IR_{in}

4 Branch to 25

25 If N=0, then branch to 1
26 offset X_{out}, Select y, Add, Z_{in}
27 Z_{out}, PC_{in}, End.

H.W= Br > 0