1. Branch Data Hazard Detection Unit:
    - the four cases of data hazards regarding branch
        - case 1

```
add $1, xx, xx
beq $1, xx, xx
```

how to solve: insert 1 stall, then refer to case 2.

- case 2

```
add $1, xx, xx
xxx xx, xx, xx   # any unrelated instruction
beq $1, xx, xx
```

how to solve: `ALUResult` forwarding from MEM stage to EX stage, through two muxes.

- case 3

```
lw  $1, xx (xx)
beq $1, xx, xx
```

how to solve: insert 1 stall to EX stage, then refer to case 4.

- case 4

```
lw  $1, xx (xx)
xx xx, xx, xx
beq $1, xx, xx
```

how to solve: insert 1 stall to EX stage.

```
input:  IDBranch, IDEXRegWrite, EXMEMRegWrite, EXMEMMemRead,
        EXDst, EXMEMDst, IDRt, IDRs;
output: forward1, forward2, PCWrite2, IFIDWrite2, Hazard2;

if ( IDEXRegWrite && IDBranch ) // case 1 and case 3
    if ( EXDst == IDRs || EXDst == IDRt )
        { PCWrite2 = 0; IFIDWrite2 = 0; Hazard2 = 1; }
if ( EXMEMRegWrite && !EXMEMMemRead && IDBranch ) // case 2
    if ( EXMEMDst == IDRs )      forward1 = 1;
    if ( EXMEMDst == IDRt )      forward2 = 1;
if ( EXMEMMemRead && IDBranch ) // case 4
    if ( EXMEMDst == IDRs || EXMEMDst == IDRt )
        { PCWrite2 = 0; IFIDWrite2 = 0; Hazard2 = 1; }
```