

# Arquitetura de Computadores

## Aulas Práticas 2021/2022

Licenciatura em Engenharia Informática e Computação  
FCUP/FEUP – Universidade do Porto

### Implementação AArch64 (III)

1. Considere que as 5 etapas do fluxo de dados (*data path*) têm as seguintes latências:

- IF: 250ps
- ID: 200ps
- EX: 250ps
- MEM: 300ps
- WB: 200ps

- (a) Qual o tempo mínimo do ciclo do relógio numa implementação da arquitetura utilizando ciclo único, multi-ciclo e *pipelining*?
- (b) Qual o tempo de execução de cada uma das seguintes instruções nessas 3 implementações (assuma um CPI ideal)?

- ADD X10, X11, X12
- LDUR X10, [X11, #40]
- STUR X10, [X11, #40]
- CBZ X10, #100
- B #200

- (c) Considere que um dado programa tem a seguinte distribuição de instruções:

- instruções do tipo R-format: 45%
- instruções do tipo load: 20%
- instruções do tipo store: 15%
- instruções de salto: 20%

Compare o tempo de execução do programa na implementação de ciclo único, multi-ciclo e *pipelining* (assuma um CPI ideal e ignore os custos associados à inicialização da *pipeline* e com *pipeline hazards*).

- (d) Se um determinado programa executa em 40 segundos na implementação de ciclo único, quanto tempo demoraria a executar na implementação com *pipelining* (assuma um CPI ideal e ignore os custos associados à inicialização da *pipeline* e com *pipeline hazards*)?

2. Considere a sequência de instruções que se segue:

```
SUB X20, X10, X11
ADD X21, X10, X12
AND X22, X21, X13
ORR X23, X21, X14
CBZ X20, X21, _label
```

Tendo por base o CPU da implementação de *pipelining* com *forwarding* apresentado na Figura 1, simule a execução das instruções acima no que diz respeito às entradas da unidade de *forwarding* e identifique os pares de instruções que requerem a utilização de *forwarding*?

3. Considere a sequência de instruções que se segue:

```
LDUR X20, [X10, #8]
ADD X21, X11, X12
AND X22, X20, X21
STUR X22, [X10, #16]
B _label
```

Tendo por base o CPU da implementação de *pipelining* com *forwarding* apresentado na Figura 1, identifique os pares de instruções em que seria necessário atrasar (*stall*) a *pipeline*?

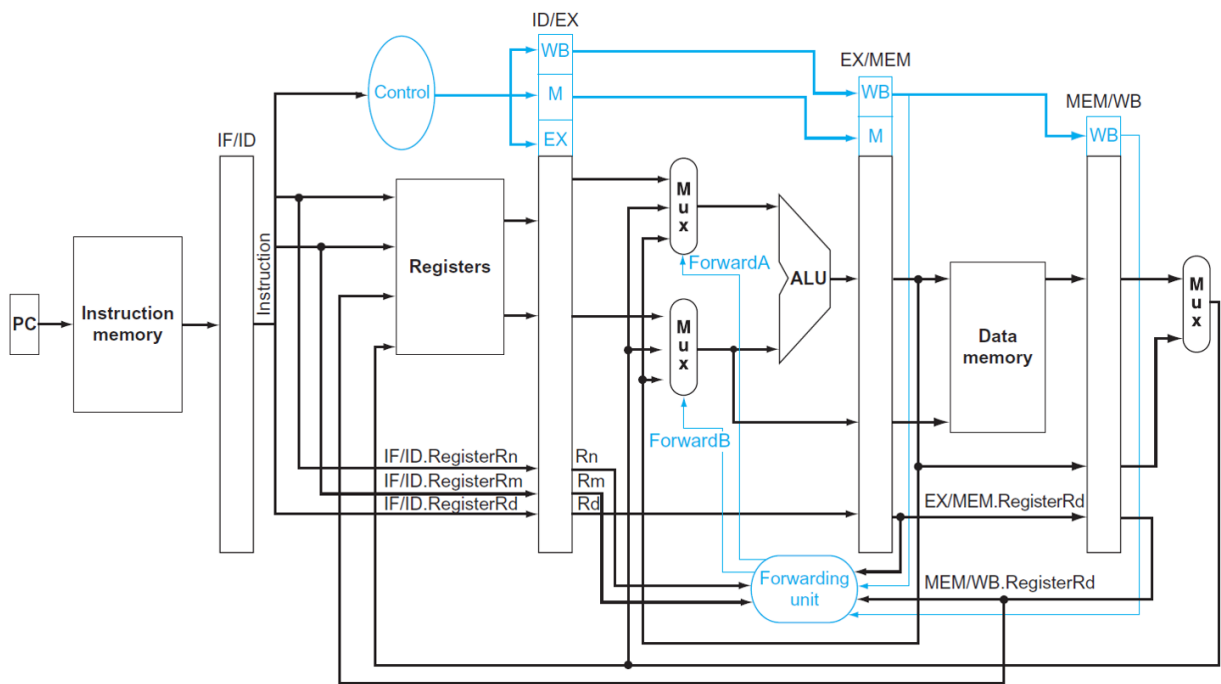


Figure 1: CPU da implementação de *pipelining* com *forwarding*