

# Pipelined Architecture (2A)

---

Copyright (c) 2014 - 2018 Young W. Lim.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Please send corrections (or suggestions) to [youngwlim@hotmail.com](mailto:youngwlim@hotmail.com).

This document was produced by using LibreOffice.

# Based on

---

ARM System-on-Chip Architecture, 2<sup>nd</sup> ed, Steve Furber

# 3-stage

fetch

- the instruction is fetched from memory
- it is placed in the instruction pipeline

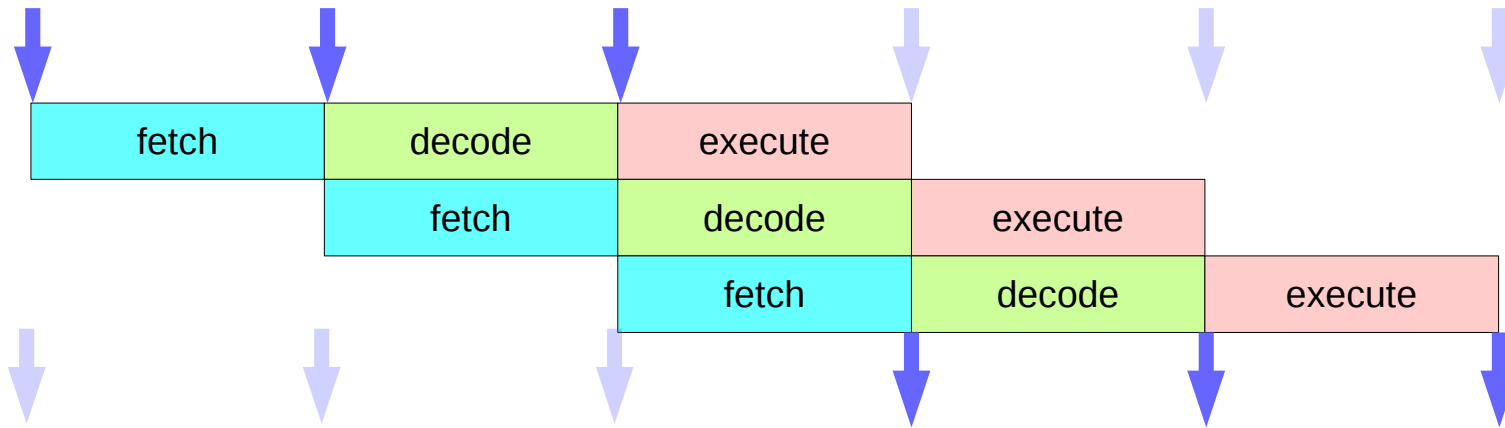
decode

- the instruction is decoded
- next cycle control signal is prepared
- the decode logic but not the datapath is dedicated

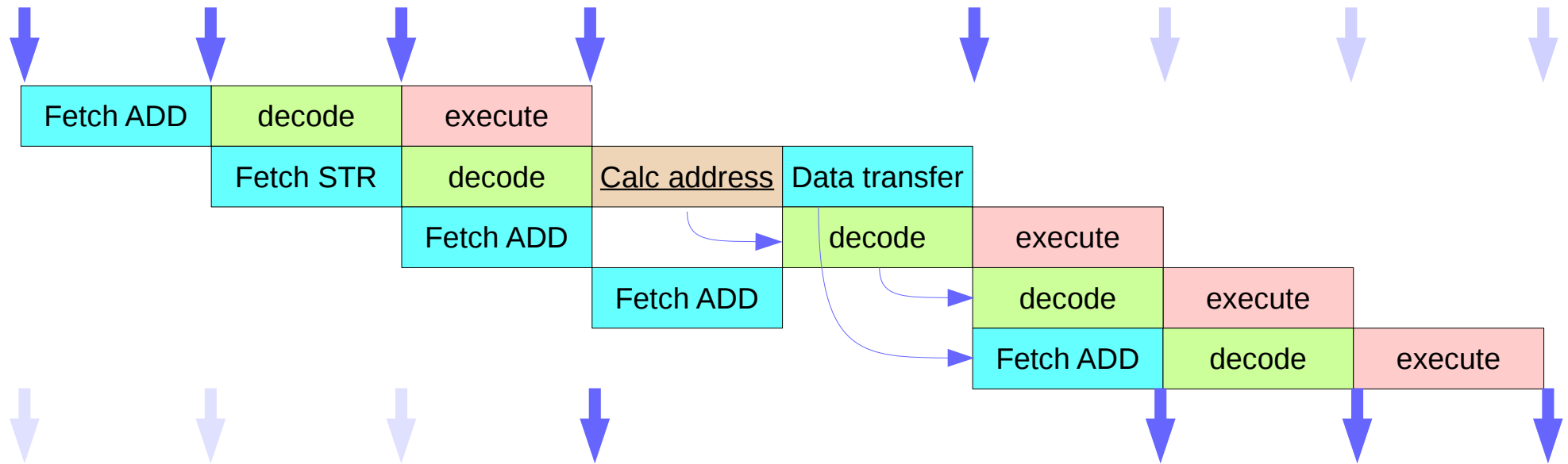
execute

- the datapath is dedicated
  - reading the register bank
  - shifting an operand
  - performing ALU operations
  - writing back the result into the register bank

# 3 stage pipeline – single cycle



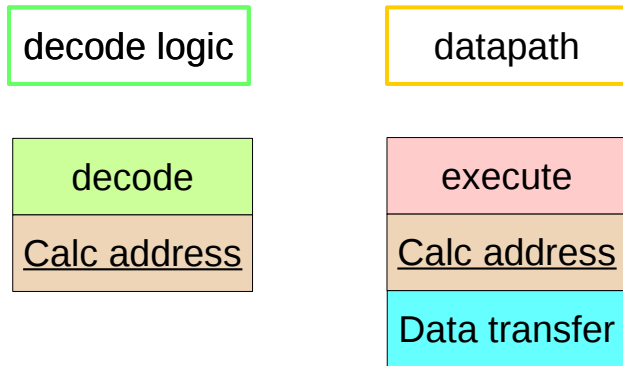
# 3-stage pipeline – multi-cycle



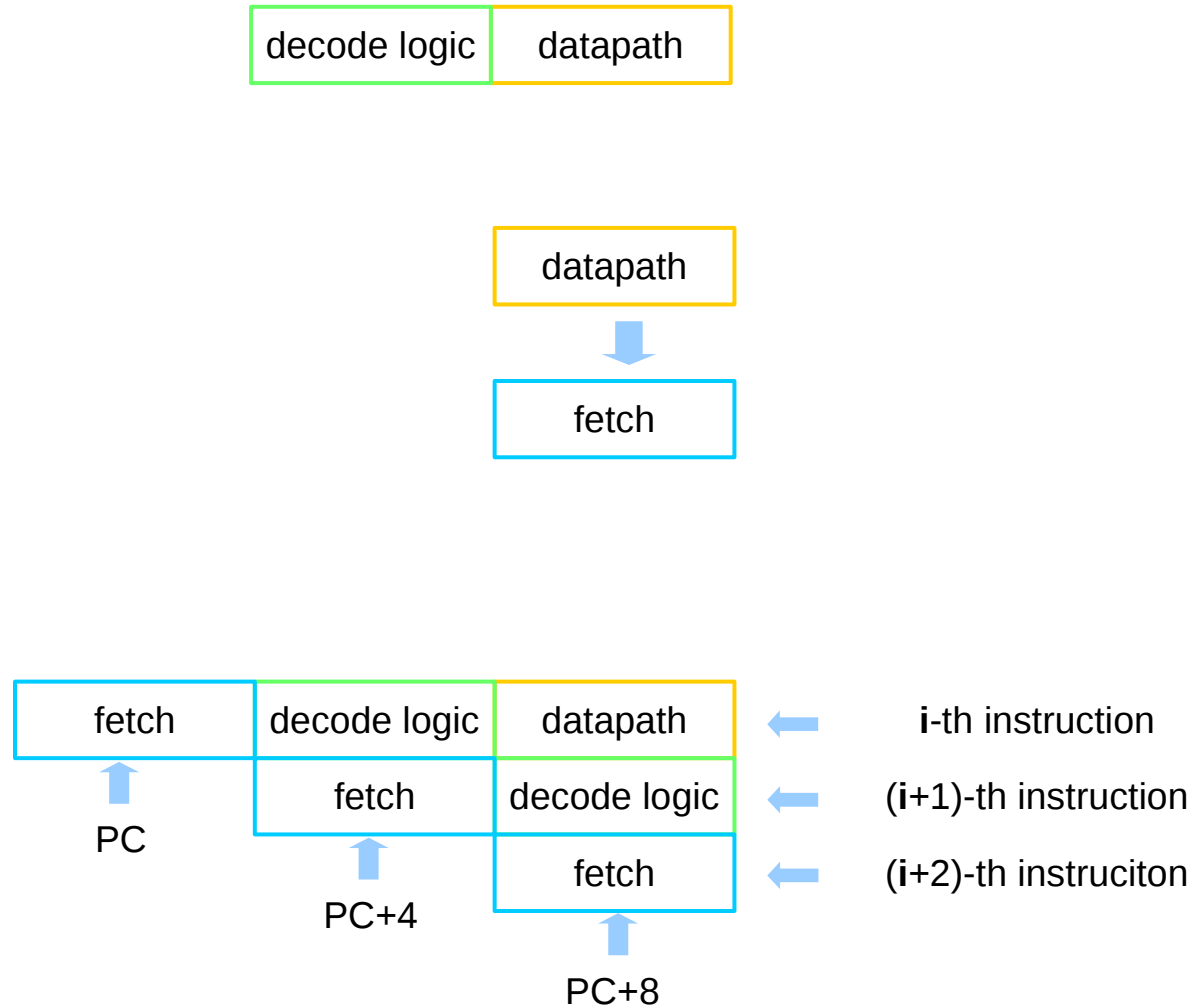
the decode logic is involved in  
all the decode cycle  
the address calculation

the datapath is involved in  
all the execute cycle  
the address calculation  
the data transfer

# 3-stage pipeline – multi-cycle

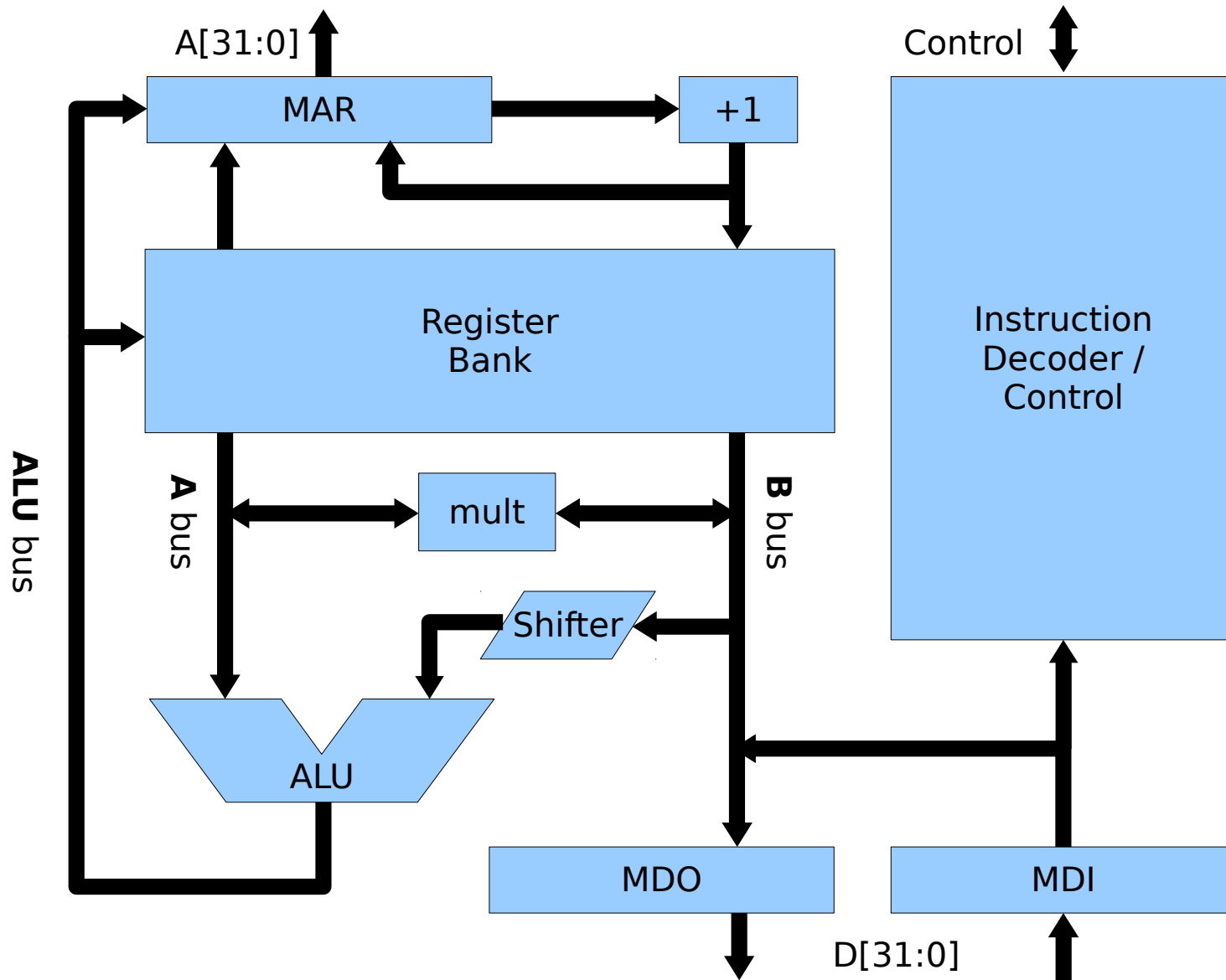


# 3-stage pipeline – multi-cycle





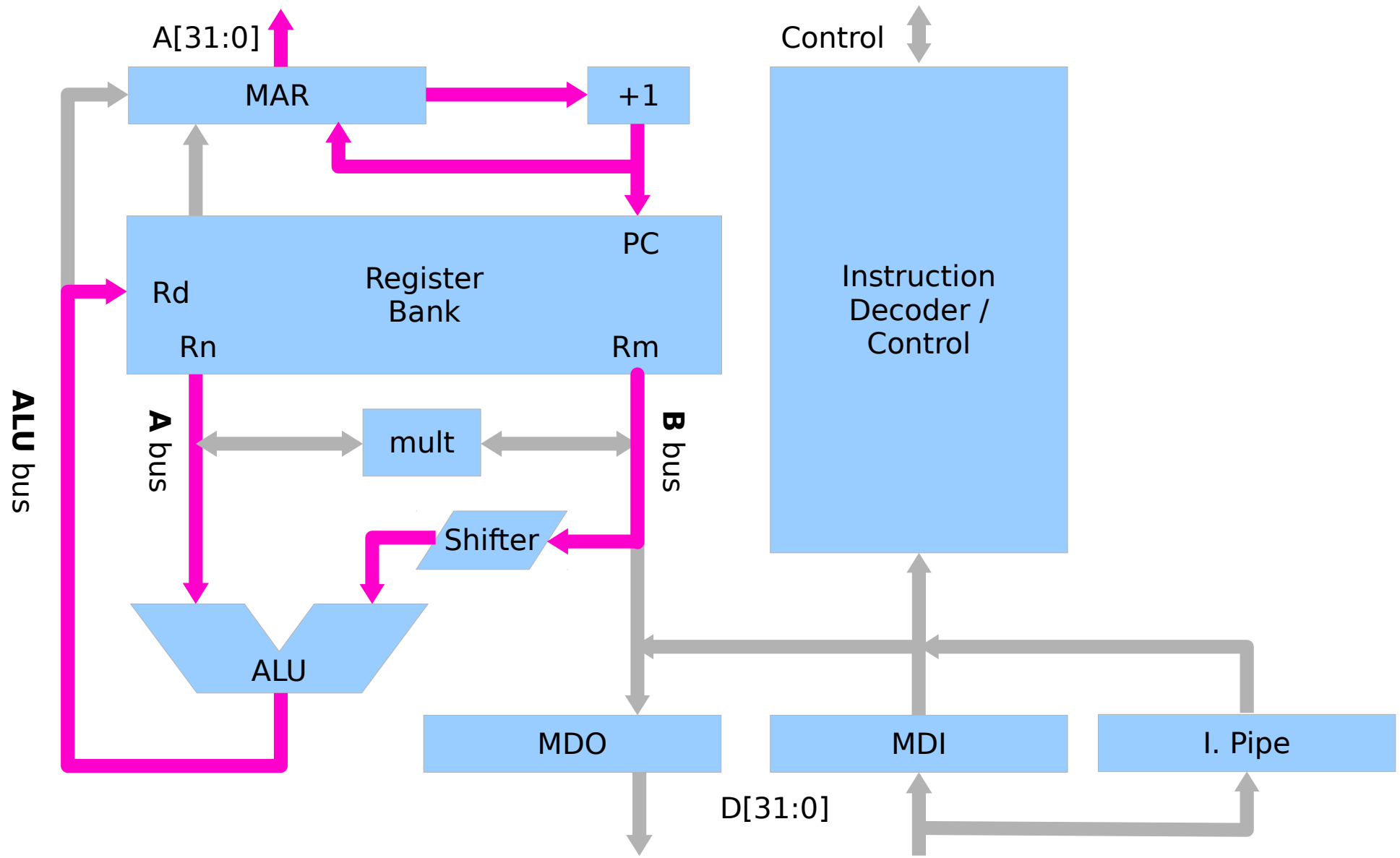
# 3-stage Pipeline



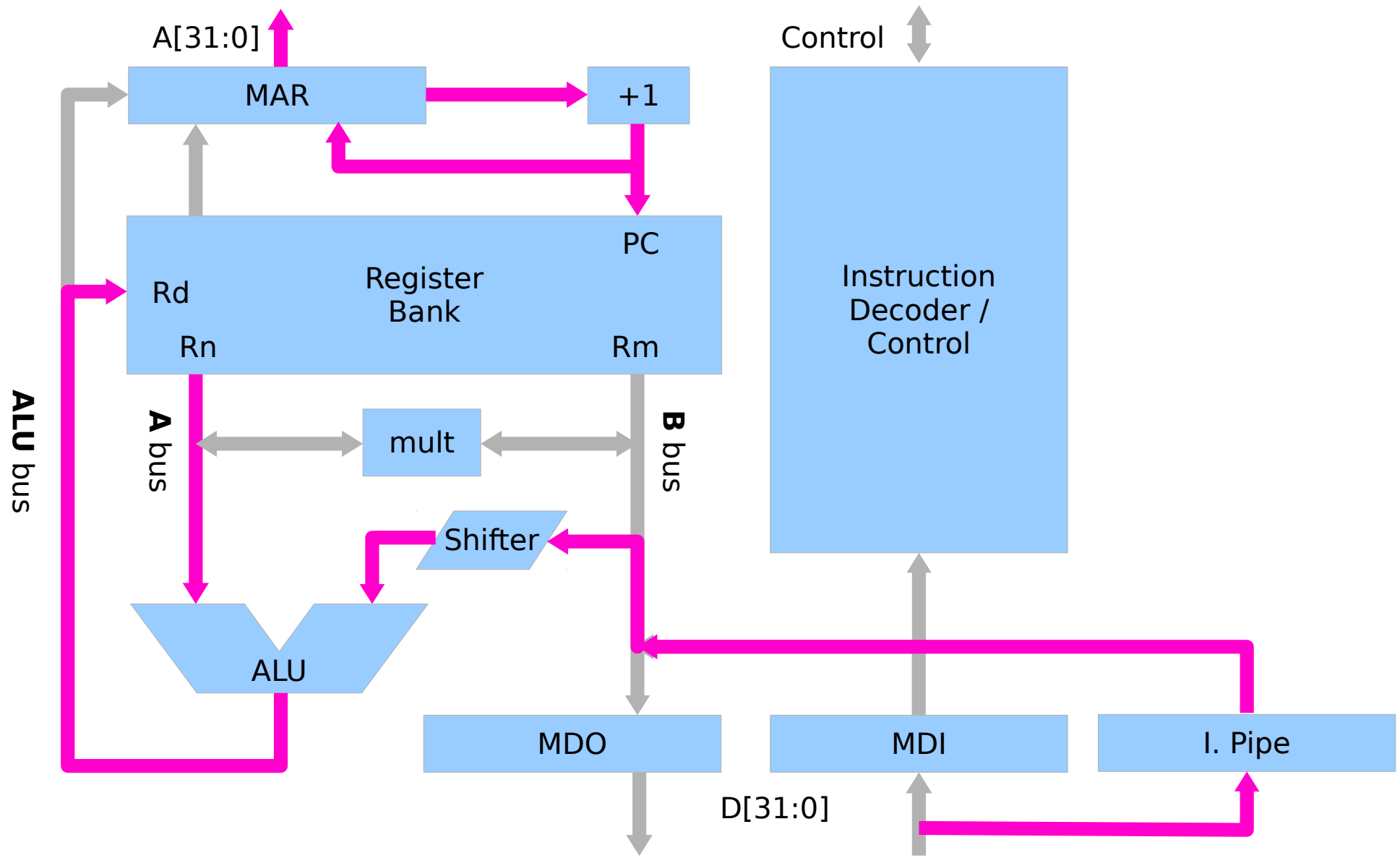
# Data Processing Instructions



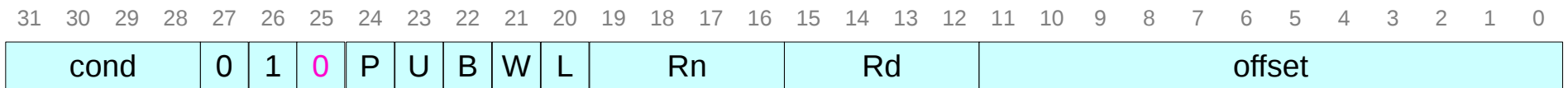
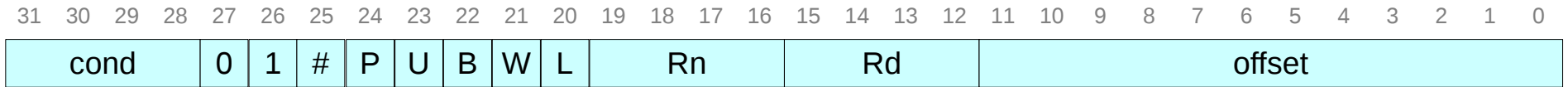
# Register-Register Operations



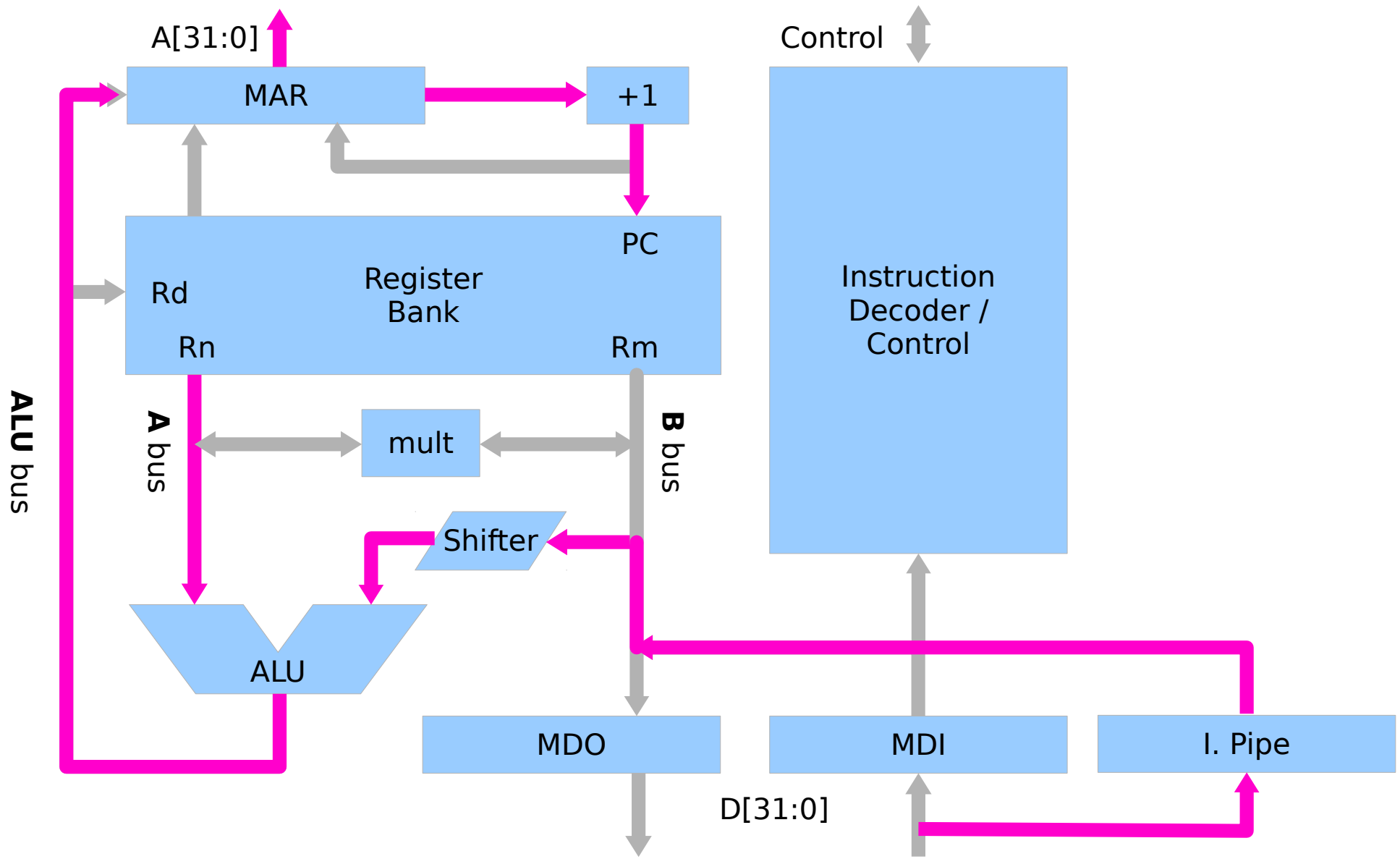
# Register-Immediate Operations



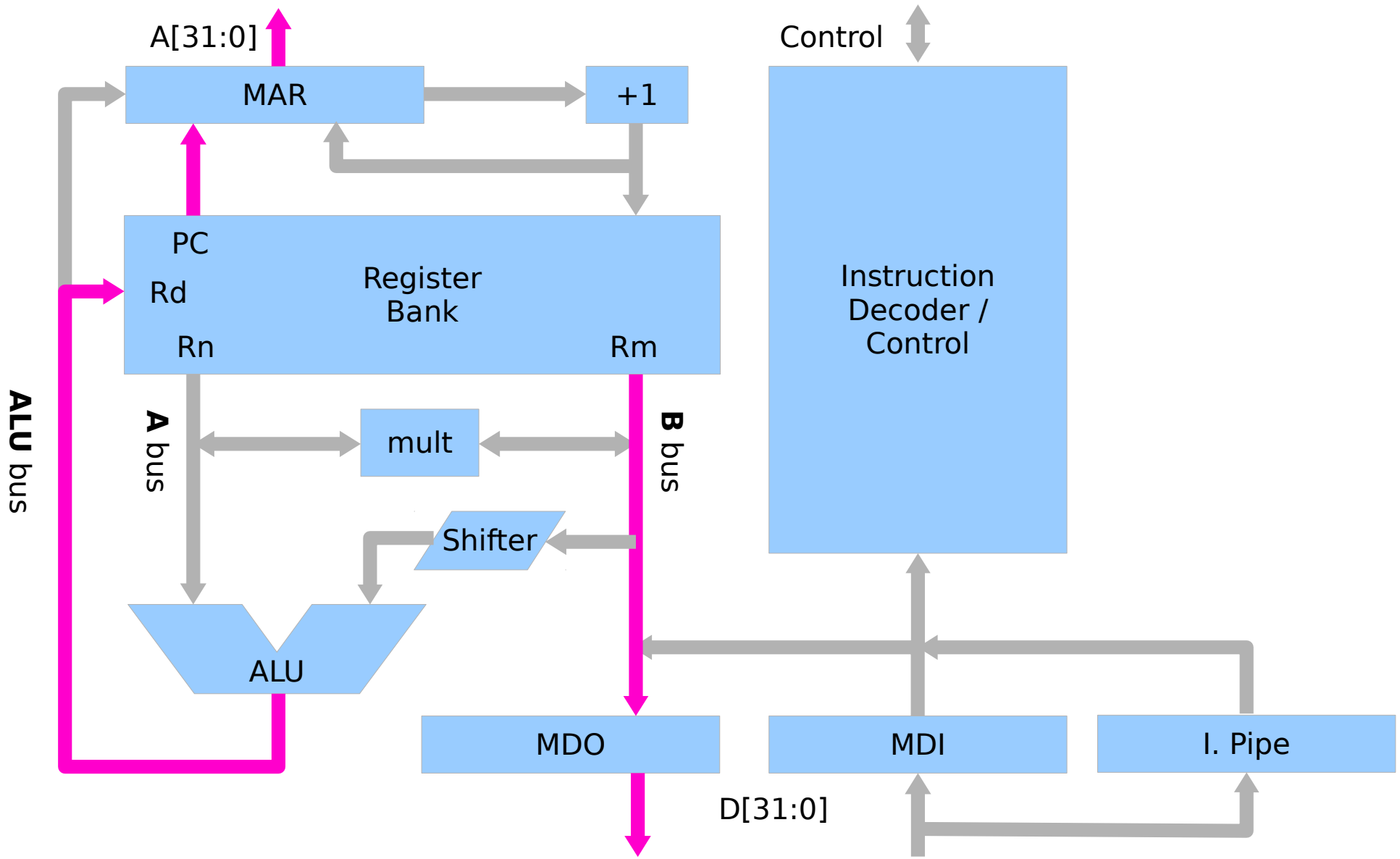
# Single word and unsigned byte data transfer instructions



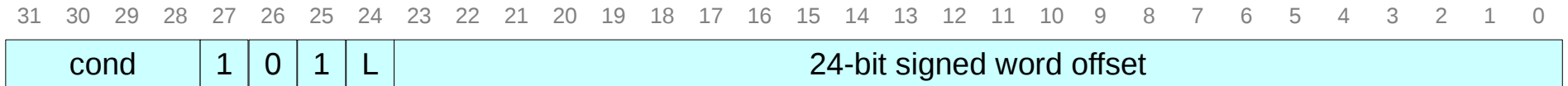
# STR - 1<sup>st</sup> Cycle



# STR - 2<sup>nd</sup> Cycle

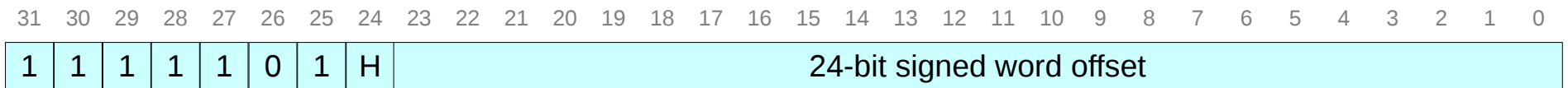
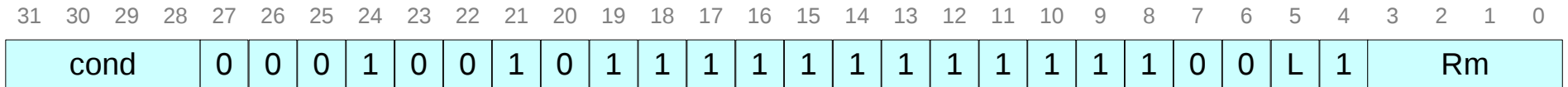


# Branch and Branch with Link (B, BL)

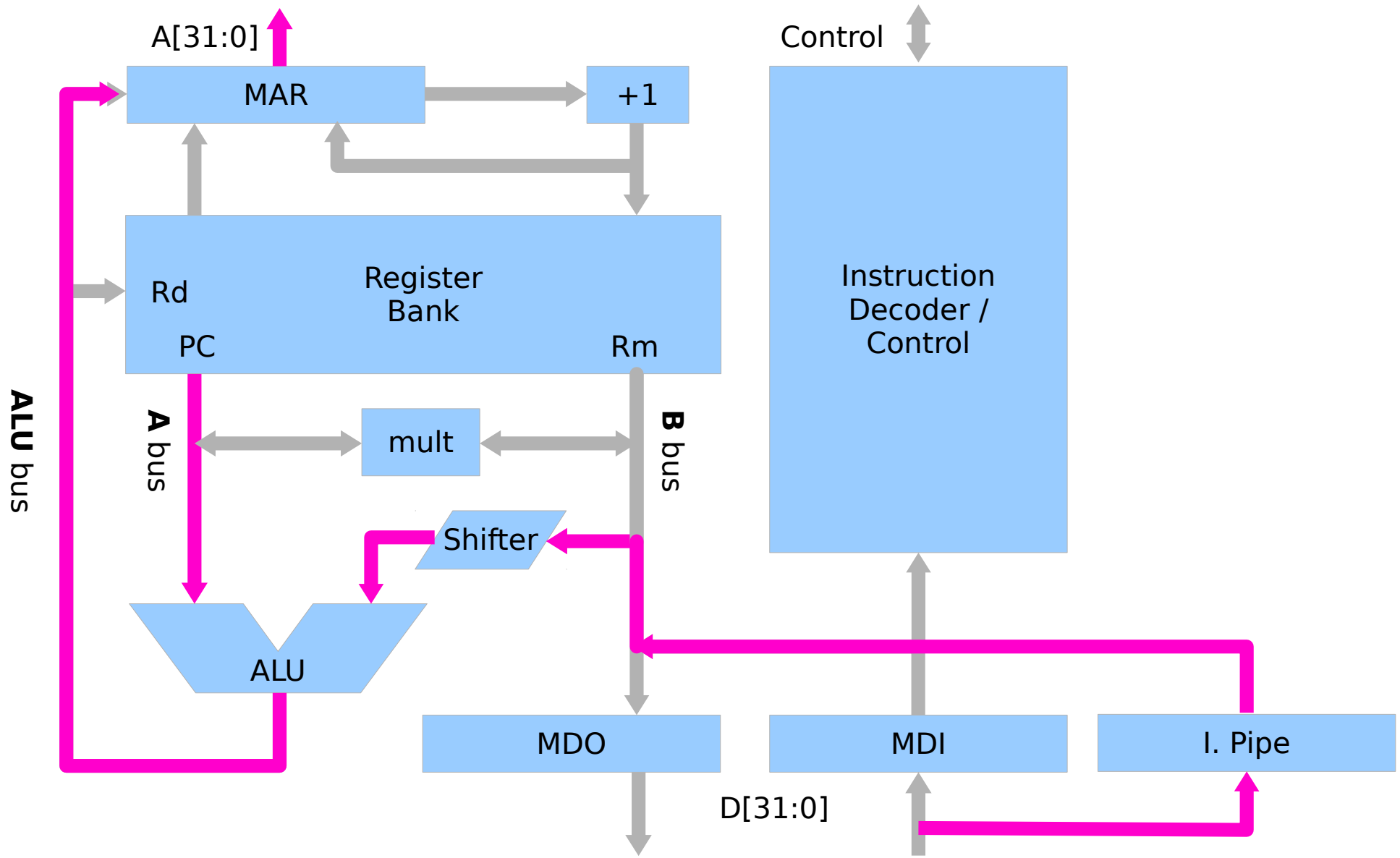




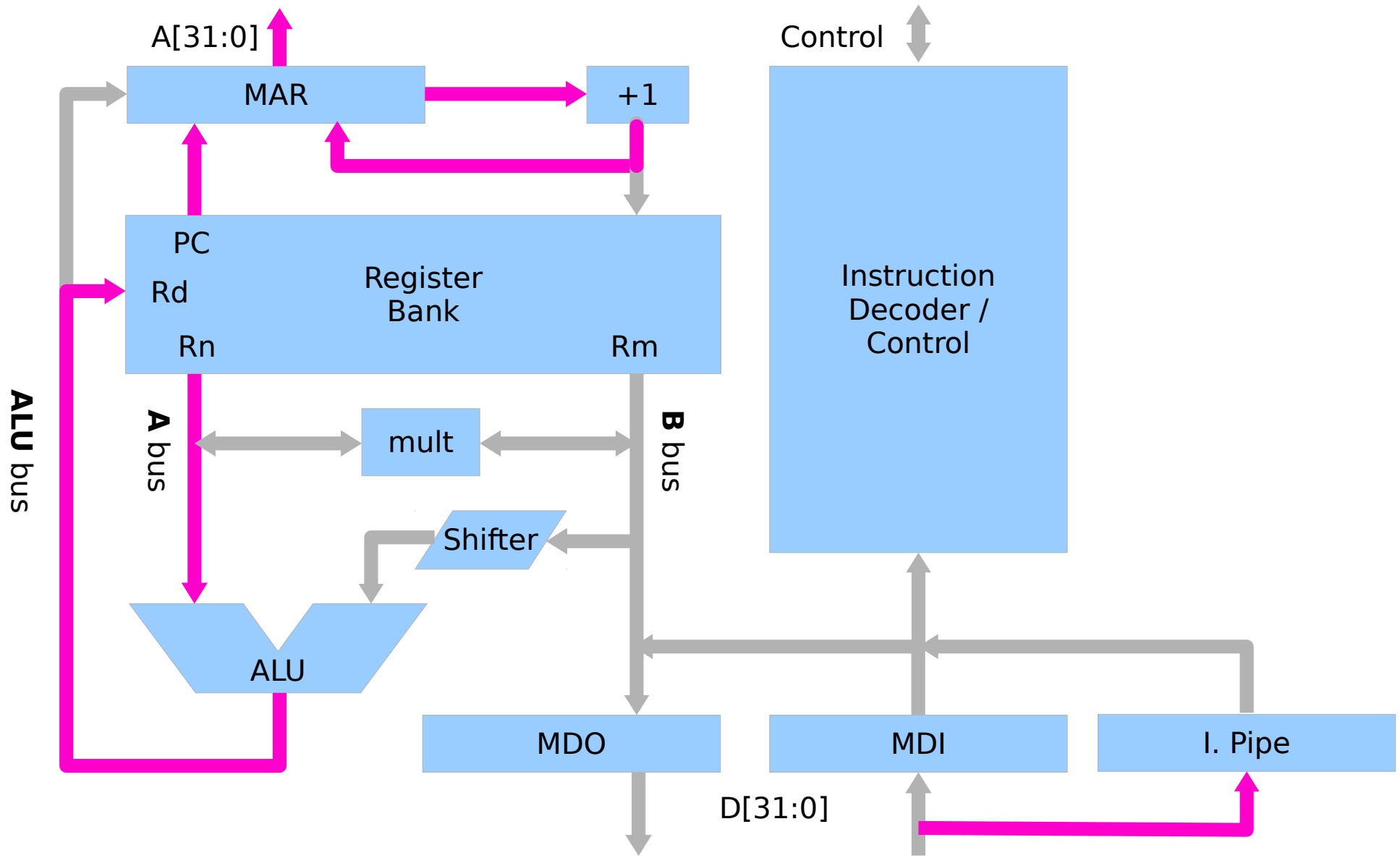
# Branch, Branch with Link and eXchange (BX, BLX)



# B - 1<sup>st</sup> Cycle



# B - 2<sup>nd</sup> Cycle



# ARM Instruction Set

---

The load-store architecture

3-address data processing instructions

(2 source registers + 1 destination register)

Conditionally executes every instruction

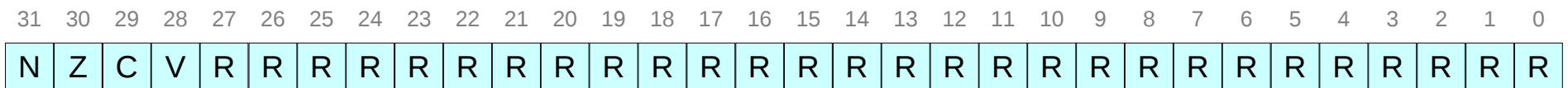
Multiple data transfer instruction

Single cycle execution of shift and ALU operations

Open instruction set for coprocessors

A very dense 16-bit compressed instruction set (Thumb)

# ARM Exception Handling



## References

- [1] <ftp://ftp.geoinfo.tuwien.ac.at/navratil/HaskellTutorial.pdf>
- [2] <https://www.umiacs.umd.edu/~hal/docs/daume02yaht.pdf>