

Computer Architecture Lab 01 – Q2

Yihang Qiu, 520030910155, 2021/11

Explanation of “q2.bin”

Use R0 and R1 to store the original bit pattern and the amount respectively. Use R5 to store the final result.

Check whether R1 is between 0 and 16 exclusively or not. If not, store 0 to x3102 and halt. (* I use x3103 as a flag, in this case store 1 to x3103.) Otherwise, store 0 to x3103.

R2 is used as the mask. Initialize R2 with 1. Continuously double R2 and decrement R1 until R1 reaches 0 (**i.e.** the only bit not masked by R2 now is the LSB after shifting R0).

Initialize R4 with 1. Double R4 and R2 and add R4 to R5 if the current bit of R0 is 1 (**i.e.** R0 AND R2=1) until R2 reaches 0 (**i.e.** overflow, **i.e.** we have scanned all bits of R1).

Store the value of R5 to x3102.

程序说明

R0 保存原序列，R1 保存移动位数，R5 保存最终结果。

检查 R1 是否合法 ($0 < R1 < 16$)，不合法则直接将 0 保存到 x3102，终止程序（注：笔者还使用 x3103 作为合法标志，此时为 1）；合法则 x3103 值为 0。

R2 初始化为 1，用作 mask。将 R2 乘 2、R1 减 1 直到 R1 为 0。此时 R2 中 1 的位置即为移位后的最低位。

R4 初始化为 1，表示移位后当前未遮盖位的权值。重复将 R4、R2 分别乘 2，R5 在 R0 当前位为 1 的时候加上 R4，直到 R2 乘 2 后溢出变为 0。

将 R5 的值保存到 x3102。