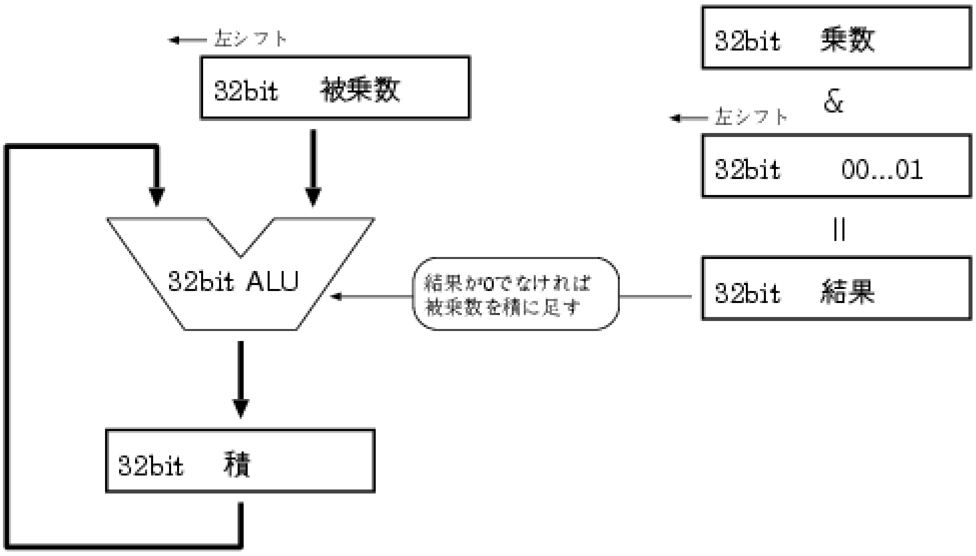
Computer Architecture (../index.html) /

Exercise 2 : MIPS Programming 1

In this exercise, we will set up a program that performs unsigned multiplication as MIPS programming and try to run it on the designed MIPS processor.

Since 1 bit left shift is equivalent to doubling, it can be easily realized by adding instruction. However, it is difficult to realize 1 bit right shift with other instructions. Therefore, this time, multiplication is realized by changing the algorithm listed in the textbook "First version of multiplication algorithm and hardware" as an algorithm that uses only left shift.

The register of multiplicand and product is 32 bits, and ALU operation can be done with 32 bits. The multiplication result to be obtained is not 64 bits, it may be only the lower 32 bits.



Details of shift realization (pershift.html)

1. Combine programs in assembly language. (Table of assembly language to use (assemble.html))
   Complete the program so that the multiplicand is initially placed at $4 (= $a0), the multiplier is at $5 (= $a1), and the product remains at $2 (= $v0). Please use $8 to $15 ($t0 ~ $t7) for temporary register.
2. Use xspim to check whether the program works properly.
   When submitting a report after first testing with a small number such as 3 × 5, the part corresponding to the period of his student ID number in the multiplicand (for example, 19 if it starts with s 119), the lower three digits of his student ID number in the multiplier Please use the value of.

About issue report

Please describe the program about the above task, attach the assembler source file, and submit the report.

- A detailed explanation of the program may be embedded as a comment in the source file.
- Please embed it in the report after emphasizing it by cutting/pasting or window image dump etc., after executing the memory displayed in xspim, enclosing the multiplied result in circle.