CLAIMS

1. A RISC-V instruction set shift instruction realization circuit, which is characterized in that the circuit is integrated with a shift module, the shift module is composed of a left shifter, a right shifter and a subtracter, when the instruction is emitted from the instruction cache, it enters the decoding module to complete the instruction decoding, at this time, the decoding information and operands enter the shift module, the subtracter is responsible for providing the shift distance Operand for the left shifter and the right shifter, and the left shifter is used to complete the left shift operation instruction, the right shifter completes the right shift operation instruction, after the operation is completed, the operation result is output, and the execution of this instruction is completed;

wherein the left shifter uses a logic expansion strobe circuit to shape the shifted data A to generate a data array tb composed of 64 data vectors *tb[i]*, wherein the data length of each vector *tb[i]* is *i*, and the generation process is as follows:

S11 sets the variable i, the initial value is 0;

S12 takes the bit of the shifted data *A[i:0]* and assigns it to *tb[i]*;

S13 inverts the data *tb[i]*;

S14 add 1 to the variable *i*, if the variable *i* is less than 64, execute S12 again, otherwise execute S15;

S15 outputs the data array *tb[i]*;

wherein the left shifter cuts the shifted data using the logic expansion method to generate a data array *sh* composed of 64 vector *sh[i]*, wherein the data width of the vector *sh[i]* is *i*, and the specific steps are as follows:

S21 sets the variable j, the initial value of *j* is 0;

S22 takes the bits considered as data*[j:0]* as *sh[j]*;

S23 adds 1 to the variable j, if the variable j is less than 64, then execute step S22 again, otherwise execute step 5;

S24 output result *sh[j]*;

S25 generates a data array *sh* consisting of 64 vectors *sh[j]*;

wherein the left shifter performs a bitwise AND operation on the nth row vector of the tb data matrix and the *sh* data matrix, respectively, where n is the row index of the *tb[i]* and *sh[j]* data arrays, at this time, the index *i* is the same as index *j*, and the result of bit by bit is logically or processed bit by bit to get the result *tmp[n]* of 1bit, since

the data matrix tb and *sh* both have 64 row vectors, the width of the vector *tmp* is 64, the process is as follows:

S31 sets the variable *n*, and the initial value of n is 0;

S32 takes *tb[n]* and *sh[n]*, and performs a bitwise AND operation on *tb[n]* and *sh[n]* to *obtain* a vector *tb_sh*;

S33 performs bitwise OR processing on the vector *tb_sh* to obtain a single-bit result *tmp[n]*;

S34 the variable n is increased by 1, if the variable n is less than 64, then execute S32 again, otherwise execute S35;

S35 output result *tmp[n]*;

the nth row vector in the *sh* data array is subjected to bit-wise logical OR operation to obtain the *sin_select[n]* vector, according to the *sin_ select[n]* bit value, the corresponding *tmp[n]* value is selected as the final shift result out *[n]*, and the output data out is also 64 bits wide.

2. The RISC-V instruction set shift instruction implementation circuit according to claim 1, wherein after determining the Operand, the shift distance Operand includes *shift_amt* and *XLEN-shift_amt*, when the instruction is a single shift operation, the shifter selects the initial shift distance data, when the instruction is cyclic shift or funnel shift, the shift operation can be divided into left shift operation and right

shift operation, and the results of left shifter and right shifter can be logically or calculated.

3. The RISC-V instruction set shift instruction implementation circuit according to claim 1, wherein the right shifter completes the right shift instruction in the same way as the left shifter completes the left shift instruction, it also uses the logic expansion strobe circuit to realize the shift function, and is the same as the shift processing unit of the left shifter, only adding the high and low inversion operation to the input and output data, when the logic and arithmetic shift to the left into 1 shift, the single-bit operation instruction result is operated by the left-to-left shifter and outputs the result, and the logic and arithmetic right shift into 1-shift instruction result is operated by the right shifter and outputs the result, finally, the result of the shift module selects the corresponding shifter data result according to the shift control signal.

4. The RISC-V instruction set shift instruction implementation circuit according to claim 1, wherein when the shift instruction is a BFP instruction, a bit region of Operand An is replaced with a piece of data from Operand B, after the OFF field and the LEN field are extracted from the script, the mask and replacement data are shifted to the left at the same time, and finally, a logical AND/OR operation is performed on the shift results of the mask and the replacement data, the specific process is as follows:

S41 selects OFF and LEN data according to the SEL field;

S42 calculates the mask based on the LEN data;

S43 uses the left shifter to shift the mask, and the shift distance is OFF data. Use the right shifter to shift the data *inb_A*, and shift the distance bit OFF data, however, the right shifter here needs to be used as a left shifter to invert the shift data *inb_A* and the output result *rishift_outb* respectively;

S44 performs logical AND/OR operation on the result;

wherein, the BFP instruction uses the pair pair mask mask data and data B [31:0] to logically shift to the left, and the right shifter is used as a left shifter, before the shift data B enters the right shift data and after the result output, the inversion operation is carried out, and the shift distance data of the left shifter and the right shifter are the same, both of which are OFF data, the left shifter data is input as mask data, and the right shifter data is input as *inb_A [31:0]*, the operation result of the BFP instruction is obtained by performing an OR/AND operation on the result.

5. The RISC-V instruction set shift instruction implementation circuit according to claim 1, wherein when the shift instruction is cyclic shift and funnel shift, the output results of the left shifter and the right

shifter are called simultaneously, and the results of the two shifters are carried out or processed;

when the shift type is cyclic left shift, the shifted data shifts n bits to the left in the left module, and the input value is zero, and the result *l_res* is obtained, in the right shift module, the shifted data is shifted to the right by 64murn bits, and the input value is zero, and the shift result *r_res* is obtained, bitwise or processed with *l_res* and *r_res*, the result of cyclic left shift is obtained, and vice versa, the left shift module shifts 64m bits to the left, and the right shift module shifts n bits to the right, the cyclic right shift result is obtained by bitwise or operation of the shift results of the two shifters.

6. The RISC-V instruction set shift instruction implementation circuit according to claim 1, wherein when the shift type is funnel shift, its operation is basically the same as the cyclic shift operation, and the shifted data of the cyclic shift data to the left and the right are the same data, while in the funnel shift operation, the left shift data and the right shift data are different data, and the data selector is responsible for data selection;

when the shift type is a single bit operation, the shifted number is 64'b1, the left shifter is used to generate mask data, and the corresponding logic operation between the operation data and the mask is completed, the single-bit operation includes the logical OR

operation, and its cyclic shift and funnel shift multiplex the operation logic in the single-bit operation.

7. The RISC-V instruction set shift instruction realization circuit according to any one of claims 1-6, wherein the processor and a memory storing execution instructions are included, when the RISC-V instruction set shift instruction implementation circuit is working, the processor executes the execution instruction stored in the memory.