**module** alu\_riscv (

**input** **logic** [**31**:**0**] a\_i,

**input** **logic** [**31**:**0**] b\_i,

**input** **logic** [**4**:**0**] alu\_op\_i,

**output** **logic** flag\_o,

**output** **logic** [**31**:**0**] result\_o

);

**logic** [**31**:**0**] add\_result;

**logic** carry;

fulladder32 adder (.a\_i(a\_i), .b\_i(b\_i), .c\_i(**0**), .s\_o(add\_result), .c\_o(carry));

// Параметры операций

**localparam** [**4**:**0**] **ADD** = **5'b00000**;

**localparam** [**4**:**0**] **SUB** = **5'b01000**;

**localparam** [**4**:**0**] **SLL** = **5'b00001**;

**localparam** [**4**:**0**] **SLTS** = **5'b00010**;

**localparam** [**4**:**0**] **SLTU** = **5'b00011**;

**localparam** [**4**:**0**] **XOR** = **5'b00100**;

**localparam** [**4**:**0**] **SRL** = **5'b00101**;

**localparam** [**4**:**0**] **SRA** = **5'b01101**;

**localparam** [**4**:**0**] **OR** = **5'b00110**;

**localparam** [**4**:**0**] **AND** = **5'b00111**;

// Вычисление результата

**always\_comb** **begin**

**case** (alu\_op\_i)

**ADD:** result\_o = add\_result;

**SUB:** result\_o = a\_i - b\_i;

**SLL:** result\_o = a\_i << b\_i[**4**:**0**];

**SLTS:** result\_o = ($**signed**(a\_i) < $**signed**(b\_i)) ? **1** : **0**;

**SLTU:** result\_o = (a\_i < b\_i) ? **1** : **0**;

**XOR:** result\_o = a\_i ^ b\_i;

**SRL:** result\_o = a\_i >> b\_i[**4**:**0**];

**SRA:** result\_o = $**signed**(a\_i) >>> b\_i[**4**:**0**];

**OR:** result\_o = a\_i | b\_i;

**AND:** result\_o = a\_i & b\_i;

**default**: result\_o = **0**;

**endcase**

**end**

// Определение флага

**always\_comb** **begin**

**case** (alu\_op\_i)

**5'b11000**: flag\_o = (a\_i == b\_i);

**5'b11001**: flag\_o = (a\_i != b\_i);

**5'b11100**: flag\_o = ($**signed**(a\_i) < $**signed**(b\_i));

**5'b11101**: flag\_o = ($**signed**(a\_i) >= $**signed**(b\_i));

**5'b11110**: flag\_o = (a\_i < b\_i);

**5'b11111**: flag\_o = (a\_i >= b\_i);

**default**: flag\_o = **0**;

**endcase**

**end**

**endmodule**



