```
    assign Rom[5'h00]=32'h2061000F;
    //001000 00011 00001 0000000000001111 addi $1,$3,F ->$1=F=15
    assign Rom[5'h01]=32'h34A2001D;
    //001101 00101 00010 0000000000011101 ori $2,$5,29 ->$2=1D=29
    assign Rom[5'h02]=32'h00221820;
    //000000 00001 00010 00011 00000 100000 add $3,$1,$2 ->$3=2C=44
    assign Rom[5'h03]=32'h00412022;
    //000000 00010 00001 00100 00000 100010 sub $4,$2,$1 ->$4=E=14
    assign Rom[5'h04]=32'h00222824;
    //000000 00001 00010 00101 00000 100100 and $5,$1,$2 ->$5=D=12
    assign Rom[5'h05]=32'h00223025;
    //000000 00001 00010 00110 00000 100101 or $6,$1,$2 ->$6=1F=31
    assign Rom[5'h06]=32'h14220003;
    //000101 00001 00010 0000000000000011 bne $1,$2,3
    //如果不相同，将跳过下面两条空指令，直接执行Rom[5'h09]指令
    assign Rom[5'h07]=32'hXXXXXXXX;
    assign Rom[5'h08]=32'hXXXXXXXX;
    assign Rom[5'h09]=32'hXXXXXXXX;
    assign Rom[5'h0A]=32'h10220002;
    //000100 00001 00010 0000000000000010 beq $1,$2,2
    //如果不相同，将不会进行跳转B
    assign Rom[5'h0B]=32'h0800000D;
    //000010 00000000000000000000001101 J 0D
    //跳转到1101 即Rom[5'h0D]指令
    assign Rom[5'h0C]=32'hXXXXXXXX;
    assign Rom[5'h0D]=32'hAD02000A;
    //101011 01000 00010 0000000000001010 sw $2 10($8) ->memory[$8+10]=$2=1D=29
    assign Rom[5'h0E]=32'h8D04000A;
 //100011 01000 00100 0000000000001010 lw $4 10($8) ->$4=memory[$8+10]=$2=1D=29
    assign Rom[5'h0F]=32'h00221826;
    //000000 00001 00010 00011 00000 100110 xor $3,$1,$2 ->$3=0x12=18
    assign Rom[5'h10]=32'h00021900;
    //000000 00000 00010 00011 00100 000000 sll $3,$2,4 ->$3=$2<<4=1D0
    assign Rom[5'h11]=32'h00021902;
    //000000 00000 00010 00011 00100 000010 srl $3,$2,4 ->$3=$2>>4=1
    assign Rom[5'h12]=32'h00021903;
    //000000 00000 00010 00011 00100 000011 sra $3,$2,4 ->$3=$2>>4=1
    assign Rom[5'h13]=32'h30470009;
    //001100 00010 00111 0000000000001001 andi $7,$2,9 ->$7=9
    assign Rom[5'h14]=32'h382F00EF;
    //001110 00001 01111 0000000011101111 xori $15,$1,0xef->$15=0xe0
    assign Rom[5'h15]=32'h3C0EF234;
    //001111 00000 00001 1111001000110100 lui $14,0xF234 ->$14=0xf234
    assign Rom[5'h16]=32'h0C00001A;
    //000011 00000 00000 0000000000011010 Jal 1A
    //执行Rom[5'h1A]指令，且$31=pc+4
    assign Rom[5'h17]=32'h0800001F;
    //000010 00000000000000000000011111 J 1F
    //跳转至Rom[5'h1F]指令
    assign Rom[5'h18]=32'hXXXXXXXX;
    assign Rom[5'h19]=32'hXXXXXXXX;
    assign Rom[5'h1A]=32'h03E00008;
    //000000 11111 00000 00000 00000 001000 Jr
    //由于$ra(31)保存了下一个指令地址,这里直接跳转到Rom[5'h17]指令
    assign Rom[5'h1B]=32'hXXXXXXXX;
    assign Rom[5'h1C]=32'hXXXXXXXX;
    assign Rom[5'h1D]=32'hXXXXXXXX;
    assign Rom[5'h1E]=32'hXXXXXXXX;
    assign Rom[5'h1F]=32'h0022F020;
    //000000 00001 00010 11110 00000 100000 add $30,$1,$2 ->$30=2C=44
```