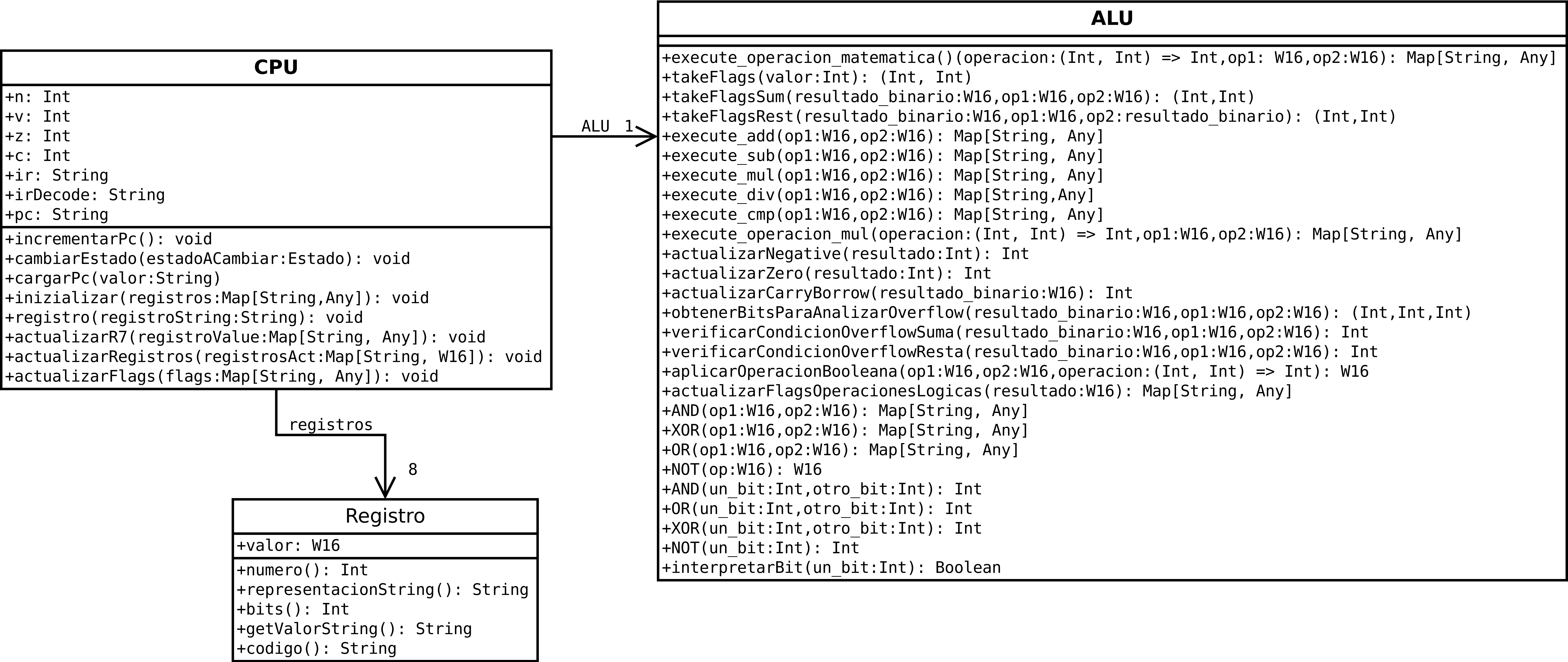
CPU
+n: Int
+v: Int
+z: Int
+c: Int
+ir: String
+irDecode: String
+pc: String
+incrementarPc(): void
+cambiarEstado(estadoACambiar:Estado): void
+cargarPc(valor:String)
+inizializar(registros:Map[String,Any]): void
+registro(registroString:String): void
+actualizarR7(registroValue:Map[String, Any]): void
+actualizarRegistros(registrosAct:Map[String, W16]): void
+actualizarFlags(flags:Map[String, Any]): void
ALU
1
ALU
+execute_operacion_matematica()(operacion:(Int, Int) => Int,op1: W16,op2:W16): Map[String, Any]
+takeFlags(valor:Int): (Int, Int)
+takeFlagsSum(resultado_binario:W16,op1:W16,op2:W16): (Int,Int)
+takeFlagsRest(resultado_binario:W16,op1:W16,op2:resultado_binario): (Int,Int)
+execute_add(op1:W16,op2:W16): Map[String, Any]
+execute_sub(op1:W16,op2:W16): Map[String, Any]
+execute_mul(op1:W16,op2:W16): Map[String, Any]
+execute_div(op1:W16,op2:W16): Map[String,Any]
+execute_cmp(op1:W16,op2:W16): Map[String, Any]
+execute_operacion_mul(operacion:(Int, Int) => Int,op1:W16,op2:W16): Map[String, Any]
+actualizarNegative(resultado:Int): Int
+actualizarZero(resultado:Int): Int
+actualizarCarryBorrow(resultado_binario:W16): Int
+obtenerBitsParaAnalizarOverflow(resultado_binario:W16,op1:W16,op2:W16): (Int,Int,Int)
+verificarCondicionOverflowSuma(resultado_binario:W16,op1:W16,op2:W16): Int
+verificarCondicionOverflowResta(resultado_binario:W16,op1:W16,op2:W16): Int
+aplicarOperacionBooleana(op1:W16,op2:W16,operacion:(Int, Int) => Int): W16
+actualizarFlagsOperacionesLogicas(resultado:W16): Map[String, Any]
+AND(op1:W16,op2:W16): Map[String, Any]
+XOR(op1:W16,op2:W16): Map[String, Any]
+OR(op1:W16,op2:W16): Map[String, Any]
+NOT(op:W16): W16
+AND(un_bit:Int,otro_bit:Int): Int
+OR(un_bit:Int,otro_bit:Int): Int
+XOR(un_bit:Int,otro_bit:Int): Int
+NOT(un_bit:Int): Int
+interpretarBit(un_bit:Int): Boolean
registros
8
Registro
+valor: W16
+numero(): Int
+representacionString(): String
+bits(): Int
+getValorString(): String
+codigo(): String