**ANÁLISIS SINTÁCTICO**

El análisis sintáctico es la segunda fase de un compilador y se encarga de generar estructuras de código, estas estructuras de código son la concatenación de componentes léxicos, por ende necesitamos los componentes léxicos generados, las estructuras que posiblemente se generen y con ello identificar y generar los posibles errores que pueden generar dichas estructuras, por ende los elementos anteriormente mencionados se definen para este compilador de la siguiente manera.

**COMPONENTES LÉXICOS DEL COMPILADOR**

Los componentes léxicos son aquellas cadenas de caracteres que concuerdan con un patrón, dicho patrón describe la forma en cómo se forma dicho componente léxico, por lo tanto los componentes léxicos de este compilador estarán formados por el lenguaje del mismo o otros componentes léxicos, y los componentes léxicos se muestran en la siguiente tabla:

|  |  |
| --- | --- |
| **COMPONENTE LÉXICO** | **PATRÓN** |
| **Número entero** | ({Digito})({Digito})\* |
| **Número real** | ({Digito})+.({Digito})({Digito})\* |
| **Condicionales** | (true | false) |
| **Identificador cadena** | ({Comilla})({Letra} | {Dígito})({Letra} | {Dígito} | {Símbolo})\*({Comilla}) |
| **Identificador** | ({Letra})({Letra} | {Digito})\* |
| **VALORES** | (**Numero Entero** | **Número Real** | **Identificador Cadena** | **Condicionales**) |
| **Puerto** | (port\_A0 | port\_A1 | port\_A2 | port\_A3 | port\_A4 | port\_A5 | port\_B0 | port\_B1 | port\_B2 | port\_B3 | port\_B4 | port\_B5 | port\_B6 | port\_B7 | port\_C2 | port\_C3 | port\_C4 | port\_C5 | port\_C6 | port\_C7 | port\_D0 | port\_D1 | port\_D2 | port\_D3 | port\_D4 | port\_D5 | port\_D6 | port\_D7) |
| **Tipo puerto** | (proximity | temperature | LED | LED\_RGB | LCD | button | motor) |
| **Tipo dato** | (int | string | decimal | boolean) |
| **Motor** | (move | restart | start) |
| **Metodo sensor** | (distance | state | time | degree) |
| **Estructura control** | (begin | loop | if | else | function) |
| **Método** | (ligther | delay | call | operation | return | print | console) |
| **Declaración** | (var | const) |
| **Operador lógico** | (and | or | not) |
| **Operador relacional** | (> | < | >= | <=) |
| **Operador aritmético** | (+ | - | \* | /) |
| **Operador asignación** | (=) |
| **Signo agrupación** | ( { | } | ( | )) |
| **signo puntuacion** | (, | .) |
| **Fin de línea** | (!) |

**ESTRUCTURAS DE CÓDIGO DEL COMPILADOR**

Las estructuras de código son generadas a través de los componentes léxicos, dónde son concatenaciones de los componentes léxicos que siguen un patrón dicha estructura, por ende las estructuras de este compilador que se pueden generar son las que están en la siguiente tabla:

|  |  |  |
| --- | --- | --- |
| **Estructuras de Declaraciones de puerto (DP)** | | |
| (Puerto) (Tipo puerto) (Identificador) (Fin de línea) | port\_A1 proximity proximitySensor! | |
| **Estructuras de Declaraciones con valor (DCV)** | | |
| (Declaración) (Tipo dato) (Identificador) (Operador asignación) (**VALORES**) (Fin de línea) | var string holaMundo = 'HolaMundo'! | |
| **Estructuras de Declaraciones sin valor (DSV)** | | |
| (Declaración) (Tipo de dato) (Identificador) (Fin de línea) | const string number3 ! | |
| **Estructura de Asignaciones (A)** | | |
| (Identificador) (Operador asignación) (**VALORES**) (Fin de línea) | number2 = 3.3 ! | |
| **Estructura de funciones de los motores con valor (FMCV)** | | |
| (Motor) (Signo agrupación “(” ) (Identificador) (Signo puntuación “,” ) (**VALORES**) (Signo puntuación “,” ) (Identificador) (Signo agrupación “)” ) (Fin De línea) | move(motor1,180,boton1)! | |
| **Estructura de funciones de los motores sin valor (FMSV)** | | |
| (Motor) (Signo agrupación “(” ) (Identificador) (Signo puntuación “,” ) (Identificador) (Signo agrupación “)” ) (Fin De línea) | start(motor1,boton1)! | |
| **Estructura de Método de impresora a consola (MIC)** | | |
| (Método “print”) (Signo agrupación “(”) (Método “console”) (Signo puntuación “,” ) (**VALORES** | Identificador) (Signo agrupación “)” ) (Fin de línea) | print(console,'El motor1 se esta moviendo')! | |
| **Estructura de Método de impresora a LCD (MIL)** | | |
| (Método “print”) (Signo agrupación “(”) (Identificador) (Signo puntuación “,” ) (**VALORES** | Identificador) (Signo agrupación “)” ) (Fin de línea) | print(Hola,'El motor1 se esta moviendo')! | |
| **Estructura de Operaciones (O)** | | |
| (Método “operation”) (Identificador) (Operador asignación) (Identificador | **VALORES**) (Operador aritmético) (Identificador | **VALORES**) (Fin de línea) | operation result = a + b! | |
| **Estructura de las funciones sin parámetros (FSP)** | | |
| (Estructura control “function”) (Identificador) (Signo agrupación “(” ) (Signo agrupación “)” ) (Signo agrupación “{” ) (**FMCV | FMSV**)\* (**MIC** | **MIL**)\* (Signo agrupación “}” ) | function moverMotor(){  move(motor1,180,boton1)!  print(console,'El motor1 se esta moviendo')!  } | |
| **Estructura de las funciones con parámetros (FCP)** | | |
| (Estructura control “function”) (Tipo dato) (Identificador) (Signo agrupación “(” ) (Tipo dato) (Identificador) ((Signos puntuación “,”) (Tipo dato) (Identificador))\* (Signo agrupación “)” ) (Signo agrupación “{” ) (**O**)\* (Método “return”) (Identificador) (Fin de línea) (Signo agrupación “}” ) | function int sumaNumerosInt(int a, int b){  operation result = a + b!  return result!  } | |
| **Estructura de llamadas de funciones con valor (LFCV)** | | |
| (Método “call”) (Identificador) (Signo agrupación “(” ) (**VALORES**) ((Signo puntuación “,”) (**VALORES**))\* (Signo agrupación “)” ) (Fin de línea) | call moverMotor(5)! | |
| **Estructura de llamadas de funciones sin valor (LFSV)** | | |
| (Método “call”) (Identificador) (Signo agrupación “(” ) (Signo agrupación “)” ) (Fin de línea) | call moverMotor()! | |
| **Estructura de control Begin (B)** | | |
| (Estructura control “begin”) (Signo agrupación “{” ) ((**LFCV**) | (**LFSV**) | (**FMCV**) | **FMSV**)\* (Signo agrupación “}” ) | begin{  call moverMotor()!  call detenerMotor()!  call escribirLCD(5)!  } | |
| **Estructura de Método de los sensores (MS)** | | |
| (Método “call”) (Identificador) (Operador asignación) (Identificador) (Signo puntuación “.” ) (Método sensor) (Signo agrupación “(” ) (Signo agrupación “)” ) (Fin de línea) | call Distancia = puerto1.distance()! | |
| **Estructura de Método Delay (MD)** | | |
| (Método “delay”) (Signo agrupación “(” ) ((Identificador) | (**VALORES**)) (Signo agrupación “)” ) (Fin de línea) | | delay(1000)! |
| **Estructura de Método Encender (ME)** | | |
| (Metodo “ligther”) (Signo agrupación “(” ) (Identificador) (Signos puntuación “,”) (Identificador) (Signo agrupación “)” ) (Fin de línea) | | ligther(led1,boton1)! |
| **Estructura de control If Relacional (IR)** | | |
| (Estructura control “if”) (Signo agrupación “(” ) ((Identificador) | (**VALORES**)) ((Operador relacional) | (Operador asignación)) ((Identificador) | (**VALORES**)) (Signo agrupación “)” ) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) | if(number1>number2){  call moverMotor()!  } | |
| **Estructura de control If Lógico con dos valores (IL2V)** | | |
| (Estructura control “if”) (Signo agrupación “(” ) ((Identificador) | (**VALORES**)) (Operador lógico) ((Identificador) | (**VALORES**)) (Signo agrupación “)” ) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) | if(number1 and number2){  call moverMotor()!  } | |
| **Estructura de control If Lógico con un valor (IL1V)** | | |
| (Estructura control “if”) (Signo agrupación “(” ) ((Identificador) | (**VALORES**)) (Signo agrupación “)” ) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) | if(number1){  call moverMotor()!  } | |
| **Estructura de control Else Relacional (ER)** | | |
| (Estructura control “if”) (Signo agrupación “(” ) ((Identificador) | (**VALORES**)) ((Operador relacional) | (Operador asignación)) ((Identificador) | (**VALORES**)) (Signo agrupación “)” ) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) (Estructura control “else”) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) | if(number1>number2){  call moverMotor()!  }else{  call detenerMotor()!  } | |
| **Estructura de control Else Lógico con dos valores (EL2V)** | | |
| (Estructura control “if”) (Signo agrupación “(” ) ((Identificador) | (**VALORES**)) (Operador lógico) ((Identificador) | (**VALORES**)) (Signo agrupación “)” ) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) (Estructura control “else”) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) | if(number1 and number2){  call moverMotor()!  }else{  call detenerMotor()!  } | |
| **Estructura de control Else Lógico con un valor (EL1V)** | | |
| (Estructura control “if”) (Signo agrupación “(” ) ((Identificador) | (**VALORES**)) (Signo agrupación “)” ) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) (Estructura control “else”) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) | if(number1){  call moverMotor()!  }else{  call detenerMotor()!  } | |
| **Estructura de control Loop (L)** | | |
| (Estructura control “loop”) (Signo agrupación “{” ) ((**MS**) | (**MIC**) | (**MIL**) | (**A**) | (**DCV**) | (**DSV**) | (**FMCV**) | (**FMSV**) | (**LFCV**) | (**LFSV** | **IR** | **IL2V** | **IL1V** | **ER** | **EL2V** | **EL1V**) | (**MD**) | (**ME**))\* (Signo agrupación “}” ) | loop{  if(number1>number2){  call moverMotor()!  }else{  call detenerMotor()!  }  call Distancia = puerto1.distance()!  number2 = 25.0 !  } | |

**ERRORES SINTÁCTICOS DEL COMPILADOR**

Los errores sintácticos son aquellas estructuras que no son aceptadas en el compilador, esto se da por no cumplir con las reglas de escritura de las estructuras, por lo tanto los errores son las concatenaciones de componentes léxicos que no son aceptados por el compilador y para este, los errores son los siguientes:

|  |  |
| --- | --- |
| **NOMBRE DEL ERROR** | **DESCRIPCIÓN** |
| Error\_03 | Falta la palabra reservada en la declaración |
| Error\_04 | Falta el tipo de puerto en la declaración |
| Error\_05 | Falta el tipo de dato en la declaración |
| Error\_06 | Falta el identificador en la declaración |
| Error\_07 | Falta el = en la declaración |
| Error\_08 | Falta el valor en la declaración |
| Error\_09 | Falta el ! en la declaración |
| Error\_10 | Faltan dos o más elementos en la declaración |
| Error\_11 | La palabra reservada no está en una declaración |
| Error\_12 | Falta el identificador en la asignación |
| Error\_13 | Falta el = en la asignación |
| Error\_14 | Falta el valor en la asignación |
| Error\_15 | Falta el ! en la asignación |
| Error\_16 | Faltan dos elementos en la asignación |
| Error\_17 | Falta la palabra reservada en la función del motor |
| Error\_18 | Falta el ( o ) o ambos en la función del motor |
| Error\_19 | Falta el identificador en la función del motor |
| Error\_20 | Falta la , en la función del motor |
| Error\_21 | Falta el valor en la función del motor |
| Error\_22 | Falta el ! en la función del motor |
| Error\_23 | Faltan dos o más elementos en la función del motor |
| Error\_24 | La palabra reservada no está en una función del motor |
| Error\_25 | Falta la palabra reservada en el método de la impresora |
| Error\_26 | Falta ( o ) o ambos en el método de la impresora |
| Error\_27 | Falta la palabra reservada console o el identificador en el método de la impresora |
| Error\_28 | Falta la , en el método de la impresora |
| Error\_29 | Falta el valor o identificador en el método de la impresora |
| Error\_30 | Falta el ! en el método de la impresora |
| Error\_31 | Faltan dos o más elementos en el método de la impresora |
| Error\_32 | La palabra reservada no está en el método de la impresora |
| Error\_33 | Falta la palabra reservada en la operación |
| Error\_34 | Falta el identificador en la operación |
| Error\_35 | Falta el = en la operación |
| Error\_36 | Falta el operando 1 en la operación |
| Error\_37 | Falta el operador aritmético en la operación |
| Error\_38 | Falta el operando 2 en la operación |
| Error\_39 | Falta el ! en la operación |
| Error\_40 | Faltan dos o más elementos en la operación |
| Error\_41 | El operador aritmético no está en una operación |
| Error\_42 | La palabra reservada no está en la operación |
| Error\_43 | Falta la palabra reservada en la función |
| Error\_44 | Falta el tipo de dato de retorno en la función |
| Error\_45 | Falta el identificador en la función |
| Error\_46 | Falta el ( o ) o ambos en la función |
| Error\_47 | Falta el tipo de dato en el o los parámetros en la función |
| Error\_48 | Falta el identificador en el o los parámetros en la función |
| Error\_49 | Falta el tipo de dato o el identificador o ambos en los parámetros en la función |
| Error\_50 | Falta la , en la función |
| Error\_51 | Falta la { o } o ambas en al función |
| Error\_52 | Falta la palabra reservada return en la función |
| Error\_53 | Falta el identificador de retorno en la función |
| Error\_54 | Falta el ! En la función |
| Error\_55 | Faltan dos o más elementos en la función |
| Error\_56 | La palabra reservada no está en la función |
| Error\_57 | Falta la palabra reservada en la llamada de la función |
| Error\_58 | Falta el identificador en la llamada de la función |
| Error\_59 | Falta el valor en la llamada de la función |
| Error\_60 | Falta el ! en la llamada de la función |
| Error\_61 | Falta el ( o ) o ambos en la llamada de la función |
| Error\_62 | Falta la , en la llamada de la función |
| Error\_63 | Falta el valor o la , en la llamada de la función |
| Error\_64 | Faltan dos o más elementos en la llamada de la función |
| Error\_65 | Falta la palabra reservada en el bloque begin o en el bloque loop |
| Error\_66 | Falta la { o } o ambas en el bloque begin |
| Error\_67 | La palabra reservada no está en el bloque begin |
| Error\_68 | Falta la palabra reservada en el método del sensor |
| Error\_69 | Falta el identificador en el método del sensor |
| Error\_70 | Falta el = en el método del sensor |
| Error\_71 | Falta el . en el método del sensor |
| Error\_72 | Falta la función del sensor en el método del sensor |
| Error\_73 | Falta el ( o ) o ambos en el método del sensor |
| Error\_74 | Falta el ! en el método del sensor |
| Error\_75 | Faltan dos o más elementos en el método del sensor |
| Error\_76 | La palabra reservada no está en la llamada de la función o en el método del sensor |
| Error\_77 | El . no está en el método del sensor |
| Error\_78 | La función del sensor no está en el método del sensor |
| Error\_79 | Falta la palabra reservada en el método delay |
| Error\_80 | Falta el ( o ) o ambos en el método delay |
| Error\_81 | Falta el valor en el método delay |
| Error\_82 | Falta el ! en el método delay |
| Error\_83 | Faltan dos o más elementos en el método delay |
| Error\_84 | La palabra reservada no está en el método delay |
| Error\_85 | Falta la palabra reservada if en el bloque if |
| Error\_86 | Falta el ( o ) o ambos en el bloque if |
| Error\_87 | Falta el valor 1 en la condición del bloque if |
| Error\_88 | Falta el operador en la condición del bloque if |
| Error\_89 | Falta el valor 2 en la condición del bloque if |
| Error\_90 | Falta la { o } o ambas del bloque if |
| Error\_91 | Falta la palabra reservada else en el bloque else |
| Error\_92 | Falta la { o } o ambas del bloque else |
| Error\_93 | Faltan dos o más elementos del bloque if |
| Error\_94 | La palabra reservada no está en el bloque if |
| Error\_95 | El operador lógico no está en el bloque if |
| Error\_96 | El operador relacional no está en el bloque if |
| Error\_97 | Falta la { o } o ambas en el bloque loop |
| Error\_98 | La palabra reservada no está en el bloque loop |
| Error\_99 | La palabra reservada no está en el método encender |
| Error\_100 | Falta el ( o ) o ambos en el método encender |
| Error\_101 | Falta el identificador en el método encender |
| Error\_102 | Falta la , en el método encender |
| Error\_103 | Falta el ! en el método encender |
| Error\_104 | Faltan dos o más elementos en el método encender |
| Error\_105 | La palabra reservada no está en el método encender |
| Error\_106 | El identificador no está en ninguna estructura |
| Error\_107 | El ! no está en ninguna estructura |
| Error\_108 | El tipo de dato no está en ninguna estructura |
| Error\_109 | El = no está en ninguna estructura |
| Error\_110 | El valor no está en ninguna estructura |
| Error\_111 | El ( no está en ninguna estructura |
| Error\_112 | El ) no está en ninguna estructura |
| Error\_113 | La , no está en ninguna estructura |
| Error\_114 | La { no está en ninguna estructura |
| Error\_115 | La } no está en ninguna estructura |
| Error\_116 | La operación no está dentro del bloque de función con parámetro |
| Error\_117 | La función del motor no está dentro de un bloque de función sin parámetros, un bloque begin, un bloque if o un bloque loop |
| Error\_118 | El método impresora no está dentro de un bloque de función sin parámetros, un bloque if o un bloque loop |
| Error\_119 | La llamada de la función no está dentro del bloque begin, un bloque if o un bloque loop |
| Error\_120 | La función del método del sensor no está dentro de un bloque if o un bloque loop |
| Error\_121 | El método delay no está dentro de un bloque if o un bloque loop |
| Error\_122 | El método encender no está dentro de un bloque if o un bloque loop |
| Error\_123 | La declaración está dentro de un bloque de función o un bloque begin |
| Error\_124 | La asignación está dentro de un bloque de función o un bloque begin |
| Error\_125 | La declaración del puerto está dentro de un bloque de función, un bloque begin, un bloque if o un bloque loop |
| Error\_126 | La función está dentro de un bloque función, un bloque begin, un bloque if o un bloque loop |
| Error\_127 | El bloque begin está dentro de un bloque de función, un bloque begin, un bloque if o un bloque loop |
| Error\_128 | El bloque if no está dentro del bloque loop |
| Error\_129 | El bloque loop está dentro de un bloque de función, un bloque begin, un bloque loop o un bloque if |
| Error\_130 | Hay varias estructuras fuera de lugar |
| Error\_XX | Hay uno o varios componentes léxicos fuera de lugar en la estructura |
| Error\_XX\_DP1 | Se esperaba un tipo de puerto en la declaración |
| Error\_XX\_DP2 | Se esperaba un identificador en la declaración |
| Error\_XX\_DP3 | Se esperaba el ! en la declaración |
| Error\_XX\_D1 | Se esperaba un tipo de dato en la declaración |
| Error\_XX\_D2 | Se esperaba un identificador en la declaración |
| Error\_XX\_D3 | Se esperaba el = en la declaración |
| Error\_XX\_D4 | Se esperaba el valor en la declaración |
| Error\_XX\_D5 | Se esperaba el ! en la declaración |
| Error\_XX\_A1 | Se esperaba el = en la asignación |
| Error\_XX\_A2 | Se esperaba el valor en la asignación |
| Error\_XX\_A3 | Se esperaba el ! en la asignación |
| Error\_XX\_FM1 | Se esperaba un ( o ) o ambos en la función del motor |
| Error\_XX\_FM2 | Se esperaba el identificador en la función del motor |
| Error\_XX\_FM3 | Se esperaba la , en la función del motor |
| Error\_XX\_FM4 | Se esperaba el valor en la función del motor |
| Error\_XX\_FM5 | Se esperaba el ! en la función del motor |
| Error\_XX\_MI1 | Se esperaba un ( o ) o ambos en el método de la impresora |
| Error\_XX\_MI2 | Se esperaba la palabra reservada console o un identificador en el método de la impresora |
| Error\_XX\_MI3 | Se esperaba la , en el método de la impresora |
| Error\_XX\_MI4 | Se esperaba el valor en el método de la impresora |
| Error\_XX\_MI5 | Se esperaba el ! en el método de la impresora |
| Error\_XX\_O1 | Se esperaba un identificador en la operación |
| Error\_XX\_O2 | Se esperaba el = en la operación |
| Error\_XX\_O3 | Se esperaba el operando 1 en la operación |
| Error\_XX\_O4 | Se esperaba el operador aritmético en la operación |
| Error\_XX\_O5 | Se esperaba el operando 2 en la operación |
| Error\_XX\_O6 | Se esperaba el ! en la operación |
| Error\_XX\_F1 | Se esperaba el tipo de dato de retorno de la función |
| Error\_XX\_F2 | Se esperaba el nombre de la función |
| Error\_XX\_F3 | Se esperaba un ( o ) o ambos en la función |
| Error\_XX\_F4 | Se esperaba la { o } o ambas en la función |
| Error\_XX\_F5 | Se esperaba el tipo de dato del parámetro de la función |
| Error\_XX\_F6 | Se esperaba el nombre del parámetro de la función |
| Error\_XX\_F7 | Se esperaba la , en la función |
| Error\_XX\_F8 | Se esperaba la palabra reservada return en la función |
| Error\_XX\_F9 | Se esperaba el identificador de retorno de la función |
| Error\_XX\_F10 | Se esperaba el ! en la función |
| Error\_XX\_LF1 | Se esperaba el identificador en la llamada de la función |
| Error\_XX\_LF2 | Se esperaba un ( o ) o ambos en la llamada de la función |
| Error\_XX\_LF3 | Se esperaba el valor en la llamada de la función |
| Error\_XX\_LF4 | Se esperaba la , en la llamada de la función |
| Error\_XX\_LF5 | Se esperaba ! en la llamada de la función |
| Error\_XX\_B1 | Se esperaba la { o } o ambas en el bloque begin |
| Error\_XX\_MS1 | Se esperaba el identificador en el método del sensor |
| Error\_XX\_MS2 | Se esperaba el = en el método del sensor |
| Error\_XX\_MS3 | Se esperaba el . en el método del sensor |
| Error\_XX\_MS4 | Se esperaba la función del sensor en el método del sensor |
| Error\_XX\_MS5 | Se esperaba un ( o ) o ambos en el método del sensor |
| Error\_XX\_MS6 | Se esperaba el ! en el método del sensor |
| Error\_XX\_MD1 | Se esperaba un ( o ) o ambos en el método delay |
| Error\_XX\_MD2 | Se esperaba un valor en el método delay |
| Error\_XX\_MD3 | Se esperaba el ! en el método delay |
| Error\_XX\_ME1 | Se esperaba un ( o ) o ambos en el método encender |
| Error\_XX\_ME2 | Se esperaba el identificador en el método encender |
| Error\_XX\_ME3 | Se esperaba la , en el método encender |
| Error\_XX\_ME4 | Se esperaba el ! en el método encender |
| Error\_XX\_I1 | Se esperaba un ( o ) o ambos en el bloque if |
| Error\_XX\_I2 | Se esperaba el valor 1 en la condición del bloque if |
| Error\_XX\_I3 | Se esperaba el valor 2 en la condición del bloque if |
| Error\_XX\_I4 | Se esperaba un operador lógico, relacional o de asignación en la condición del bloque if |
| Error\_XX\_I5 | Se esperaba la { o } o ambas en el bloque if |
| Error\_XX\_I6 | Se esperaba la palabra reservada else en el bloque if |
| Error\_XX\_I7 | Se esperaba la { o } o ambas en el bloque else |
| Error\_XX\_L1 | Se esperaba la { o } o ambas en el bloque loop |